Ensemble neural network approach detecting pain intensity from facial expressions

Manuscript Details

Manuscript number	AIIM_2020_86_R3
Title	Ensemble neural network approach detecting pain intensity from facial expressions
Article type	Research Paper

Abstract

This paper reports on research to design an ensemble deep learning framework that integrates fine-tuned, three-stream hybrid deep neural network (i.e., Ensemble Deep Learning Model, EDLM), employing Convolutional Neural Network (CNN) to extract facial image features, detect and accurately classify the pain. To develop the approach, the VGGFace is fine-tuned and integrated with Principal Component Analysis and employed to extract features in images from the Multimodal Intensity Pain database at the early phase of the model fusion. Subsequently, a late fusion, three layers hybrid CNN and recurrent neural network algorithm is developed with their outputs merged to produce image-classified features to classify pain levels. The EDLM model is then benchmarked by means of a single-stream deep learning model including several competing models based on deep learning methods. The results obtained indicate that the proposed framework is able to outperform the competing methods, applied in a multi-level pain detection database to produce a feature classification accuracy that exceeds 89%, with a receiver operating characteristic of 93%. To evaluate the generalization of the proposed EDLM model, the UNBC-McMaster Shoulder Pain dataset is used as a test dataset for all of the modelling experiments, which reveals the efficacy of the proposed method for pain classification from facial images. The study concludes that the proposed EDLM model can accurately classify pain and generate multi-class pain levels for potential applications in the medical informatics area, and should therefore, be explored further in expert systems for detecting and classifying the pain intensity of patients, and automatically evaluating the patients' pain level accurately.

Keywords	ensemble neural network; pain detection; facial expression; deep learning
Manuscript region of origin	Asia Pacific

Corresponding Author	Ghazal Bargshady
Corresponding Author's Institution	University of Southern Queensland (USQ)
Order of Authors	Ghazal Bargshady, Xujuan Zhou, Ravinesh Deo, Jeffrey Soar, Frank Whittaker, Hua Wang

Submission Files Included in this PDF

File Name [File Type]

Title Page.docx [Cover Letter]

response letter - R2.docx [Response to Reviewers]

Highlightes.docx [Highlights]

revised-manuscript - R2.docx [Manuscript File]

Figures - R2.docx [Figure]

Tables - R2.docx [Table]

declaration-of-competing-interests.docx [Conflict of Interest]

To view all the submission files, including those not included in the PDF, click on the manuscript title on your EVISE Homepage, then click 'Download zip file'.

Research Data Related to this Submission

There are no linked research data sets for this submission. The following reason is given:
The authors do not have permission to share data

An ensemble neural network approach to detect pain intensity from facial expressions

Ghazal Bargshady (ghazal.bargshady@usq.edu.au)[a*], Xujuan Zhou (xujuan.zhou@usq.edu.au)[a], Ravinesh C Deo (ravinesh.deo@usq.edu.au)[b], Jeffrey Soar (jeffrey.soar@usq.edu.au)[a], Frank Whittaker(Frank.Whittaker@usq.edu.au)[a], Hua Wang(Hua.Wang@vu.edu.au)[c]

[a] School of Management and Enterprise
University of Southern Queensland
Springfield, Qld 4300, Australia

[b] School of Sciences
University of Southern Queensland
Springfield, Qld 4300, Australia

[c] Victoria University, Melbourne, Australia

[*]Corresponding author: Ghazal Bargshady, email: ghazal.bargshady@usq.edu.au, phone: (+61)(405)(415-178), address: 37 Sinnathamby Blvd, Springfield Central QLD 4300, University of Southern Queensland (USQ) Springfield campus

AIIM: Editorial Decision on AIIM_2020_86_R2

An ensemble neural network approach to detect pain intensity from facial expressions

To

Professor Carlo Combi

Editor-in-Chief for Artificial Intelligence in Medicine

Re: Revised Manuscript AIIM_2020_86_R2

Dear Prof Combi

The authors would like to express their gratitude to reviewers and the journal editorial board.

The authors also thank you for the opportunity to revise the manuscript.

The authors have carefully considered all the comments. In response, we have revised the paper carefully, ensuring the reviewers' comments are addressed properly. The revisions made to this paper can be tracked in the revised manuscript as they are written in green font.

In the revised document, we also provide our response all the reviewers' comments, which is presented in the table below. The left side shows the comments from the reviewers and the right-hand side shows the author responses demonstrating how the necessary revisions were made.

Reviewer_1

	Reviewer Comment(s)	**Author Responses**
1	In Fig.6, you have to present the same person for each level.	Done. Figure 6 has been updated as the review mentioned by a same person for each level. Please see line 303.
2	In Fig. 7, also you have to present the same person for each level. Actually, In Fig. 6 and Fig.7 some images are looks like reused for different level image.	Done. Figure 7 has been updated as the review mentioned by a same person for each level. Please see line 314.

Reviewer_2

	Reviewer Comment(s)	**Author Responses**
1	Line 646: References are re-indexed from 1.	Thanks for picking this issue. References issue has been fixed.
2	Line 214, 223, 225, 235 should start with lower case character.	Done. Please see the lines 211, 220, 222, 232.
3	Line 221: what is X []?	Corrected. Please see line 218.
4	Line 274: should move the table caption to next page	Done. Please see line 271.
5	Line 280: should be " Experimental configuration and database**s**"?	Fixed. Please see line 277.
6	Line 288: should be "The MIntPAIN database"?	Fixed. Please see line 285.

The authors wish to thank you for multiple rounds of revisions of our paper.
We hope now the paper can be accepted for publication.
Sincerely
Ghazal Bargshady, 20/08/2020
On behalf of all co-authors

Highlights:

- Automated detection of pain from facial expressions is a challenge in medical care.
- A new ensemble deep neural network algorithm designed to improve automatic pain detection.
- The performance of the new proposed ensemble deep learning algorithm for detecting pain in 5 level is high and tested in two different databases.

Ensemble neural network approach detecting pain intensity from facial expressions

Ghazal Bargshady (ghazal.bargshady@usq.edu.au)[a*], Xujuan Zhou (xujuan.zhou@usq.edu.au)[a], Ravinesh C Deo (ravinesh.deo@usq.edu.au)[b], Jeffrey Soar (jeffrey.soar@usq.edu.au)[a], Frank Whittaker(Frank.Whittaker@usq.edu.au)[a], Hua Wang(Hua.Wang@vu.edu.au)[c]

[a] School of Management and Enterprise
University of Southern Queensland
Springfield, Qld 4300, Australia

[b] School of Sciences
University of Southern Queensland
Springfield, Qld 4300, Australia

[c] Victoria University, Melbourne, Australia

[*]Corresponding author: Ghazal Bargshady, email: ghazal.bargshady@usq.edu.au, phone: (+61)(405)(415-178), address: 37 Sinnathamby Blvd, Springfield Central QLD 4300, University of Southern Queensland (USQ) Springfield campus

Abstract

This paper reports on research to design an ensemble deep learning framework that integrates fine-tuned, three-stream hybrid deep neural network (*i.e.*, Ensemble Deep Learning Model, EDLM), employing Convolutional Neural Network (CNN) to extract facial image features, detect and accurately classify the pain. To develop the approach, the VGGFace is fine-tuned and integrated with Principal Component Analysis and employed to extract features in images from the Multimodal Intensity Pain database at the early phase of the model fusion. Subsequently, a late fusion, three layers hybrid CNN and recurrent neural network algorithm is developed with their outputs merged to produce image-classified features to classify pain levels. The EDLM model is then benchmarked by means of a single-stream deep learning model including several competing models based on deep learning methods. The results obtained indicate that the proposed framework is able to outperform the competing methods, applied in a multi-level pain detection database to produce a feature classification accuracy that exceeds 89%, with a receiver operating characteristic of 93%. To evaluate the generalization of the proposed EDLM model, the UNBC-McMaster Shoulder Pain dataset is used as a test dataset for all of the modelling experiments, which reveals the efficacy of the proposed method for pain classification from facial images. The study concludes that the proposed EDLM model can accurately classify pain and generate multi-class pain levels for potential applications in the medical informatics area, and should therefore, be explored further in expert systems for detecting and classifying the pain intensity of patients, and automatically evaluating the patients' pain level accurately.

Keywords: ensemble neural network, pain detection, facial expression, deep learning

1. Introduction

Pain is a significant indicator of human discomfort and an indicator of the need for medical diagnosis of a possible disease and its related treatments in patients. It is usually measured by clinicians, albeit, employing largely a manual approach such as using a self-reported pain detection system. Various pain measurement scales have been designed to describe a patient's self-report of pain intensity, including but not limited to the Visual Analogue Scale (VAS)[1], Verbal Rating Scale (VRS), Faces Pain Scale-Revised (FPSR), and the Numerical Rating Scale (NRS) [1]. However, self-reported pain level assessment may not always be the appropriate method for different disease contexts and patients' scenarios [1, 2]. Moreover, in doing so, this task may require greater intellectual and dialectal abilities that makes the self-reporting impractical for populations such infants and elderly patients lacking effective communication skills [3, 4]. An automated decision support system for pain assessment that utilises facial image processing can provide an effective alternative medium to the self-reporting method to more accurately evaluate the severity of pain. Two examples of such systems include the Facial Action Coding System (FACS) [5] and the Prkachin and Solomon Pain Intensity (PSPI) scale [6]. However, automatically assessing the pain level from facial images or video recordings can be a challenging task because of the presence of several external and complicating factors (*e.g.*, phenomenon of human smiles in spite of pain and the gender related pain tolerating abilities [7]). This means that we are likely to face a major challenge in terms of accurate facial expression recognition and interpretation due to the relatively large visual features with considerable variation caused by person-to-person characteristics, their expressions and the variations in face appearance caused by many extrinsic conditions such as illumination and the point of view [8]. Another key challenge in facial expression recognition arises from the need to develop effective representation that balance the complex distribution of intra- and inter- class variations [9]. Effective methods that demarcate true facial features associated with a pain level and the causal factor (*i.e.* medical condition) are highly warranted to support rapidly evolving medical informatics capabilities.

[1] *List of acronyms*

AAM (Active Appearance Models)
ARC (Australian Research Council)
ASM (Active Shape Model)
AUC (Area under Curve)
BiLSTM (Bidirectional Long Short Memory)
CNN (Convolutional Neural Network)
D (Depth)
EDLM (Ensemble Deep Learning Model)
FACS (Facial Action Coding Systems)
FN (False Negative)
FP (False Positive)
FPR (False Positive Rate)
FPSR (Faces Pain Scale-Revised)
LBP (Local Binary Pattern)
LSTM (Long Short-Term Memory)
MAE (Mean Absolute Error)
MIntPAIN (Multimodal Intensity Pain)
MSE (Mean Squared Error)
NRS (Numerical Rating Scale)
PCA (Principal Components Analysis)
RNN (Recurrent Neural Network)
PSPI (Prkachin and Solomon Pain Intensity)
ROC (Receiver Operating Characteristics)
SVM (Support Vector Machine)
TN (True Negative)
TP (True Positive)
TPR (True Positive Rate)
T (Thermal)
VRS (Verbal Rating Scale)
VAS (Visual Analogue Scale)

Artificial intelligence (AI) algorithms in an automatic pain detection system that analyse concealed features using indicators of pain (*e.g.*, a face image) can potentially provide medical practitioners a more intelligent approach to investigate the actual pain level prior to treating the relevant disease. Recently, deep learning methods employing multiple hidden layer neuronal systems for feature extraction have gained importance as a mainstream automated technique for this purpose, with its increasing capacities to perform complex and highly nonlinear predictive modelling tasks (*e.g.*, classification and feature extraction) from relatively complex datasets such as human face images that indicate a medical condition. Many deep learning techniques, including convolutional neural networks (CNN) [10], and recurrent neural networks (RNN) [11], have thus been explored for facial expression analysis and pain detection.

In spite of many AI methods tested for feature extraction, the ensemble-based approaches where two or more algorithms are integrated to capture the merits of each for improved accuracy is being widely developed for multi-purpose classification tasks [12, 13]. The popularity of ensemble-based methods is perhaps attributable to their relatively superior performance in comparison to the other single deep learning algorithms. The study of [14] provided three important reasons to adopt this method, including their statistical, computational, and representational efficacy compared to single algorithm learning models. Indeed, increasing the number of stacked hidden layers and neuronal networks depth can improve the clarity of features learned from the CNNs and, and therefore, improving the performance of deep neural networks in image processing tasks [15]. Therefore, in this research paper we aim to build and test a new ensemble deep learning model to recognise the multi-classification level pain intensity employing the patient's video frame images.

The proposed ensemble model consists of two steps including future extraction as early fusion and classification as late fusion. In the early fusion section, a newly developed feature extraction technique has been applied based on the fine-tuned VGGFace algorithm that integrates Principal Component Analysis (PCA) hieratically to extract the features embedded in human face images. Henceforth, in the late fusion section, a three-stream CNN-RNN network has been designed, and finally, the resulting facial image features are merged as the output of the ensemble classification model. The proposed algorithm is tested comprehensively by employing two unique databases. First the Multimodal Intensity Pain (MIntPAIN) database [2, 16] with labelled video sequences in terms of the VAS metric and second, the UNBC-McMaster Shoulder Pain Archive Dataset [17] with labelled video frames in terms of PSPI and FACS metrics are used.

More precisely, the novelty of this research is as follows:

1) A new image classification approach with an early fusion section is constructed for effective feature extraction by adopting the fine-tuning VGG-Face pre-trainer, and its outputs are

integrated with PCA to extract the features more effectively and efficiency by reducing the dimensionality of image dataset.

2) A new image classification approach that includes a three-stream ensemble CNN-RNN classifier system, where the outputs are merged by means of the late fusion section to finally classify the pain level in five distinct levels, resulting from the extracted features from human facial images.

3) The overall framework denoted as Ensemble Deep Learning Model (EDLM) model is trained and tested utilising two popular face databases represented with various pain features and the obtained results are used to benchmark EDLM against state-of-the-art techniques as the baseline model.

The rest of the paper is organized as follows: In Section 2, the existing methods and related works are described. In Section 3, an overview of the proposed EDLM model is introduced. Next, the experiments and databases are presented in Section 4 while the results and discussions are provided in Section 5, with conclusions and future works outlined in Section 6.

2. Related works

In the following, the related studies in pain detection from facial expressions, including a general overview of deep learning techniques, existing research and ensemble neural networks are described.

2.1 Deep learning used in facial expressions

CNNs have been used to image classification and applied to identify face and objects effectively [18, 19]. CNNs and their pre-trained algorithms obtained notable results especially in image classification and feature extraction [20]. In addition, recently CNNs models have achieved higher performances on the ImageNet dataset such as AlexNet [10], GoogLeNet [21]. Features extracted from pre-trained CNNs used in computer vision tasks such as emotion recognition and object detection and the achieve results indicated better performance in comparison with handcrafted features.

Even though deep learning methods are powerful tools for tasks estimation, however; they are not suitable for analysing sequential data such as speech or video data. Therefore, RNN was designed to represent features in capturing information from all the earlier time steps and to renew its representation through upcoming information [22]. Long short-term memory (LSTM) deep learning is based on RNN architecture and unlike feedforward neural networks it has feedback connection. Standard RNNs can learn based on long-term dependencies like LSTM but training them is difficult since the gradients tend to vanish or explode. LSTM has a cell state under control by three gates as: forget, input, and output gates. The Forget gate keeps relevant information from prior steps. The input gate adds relevant information from the current step. The output gate determines the next hidden state status [23, 24]. Fig. 1 shows the architecture of an LSTM cell, in which the cell state part is calculated by:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{1}$$

The output of the forget gate is calculated as:

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{2}$$

The cell state for the current time-step is calculated as following:

$$c_t = f_t c_{t-1} + i_t tanh(W_{xc}x_t + w_{hc}h_{t-1} + b_c) \tag{3}$$

Once the forget and input gates have controlled the amount of information in the earlier cell state c_{t-1} and the new cell state c_t should be let through.

The state can expect the output of the cell as following:

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_o c_t + b_o) \tag{4}$$

$$h_t = o_t tanh(c_t) \tag{5}$$

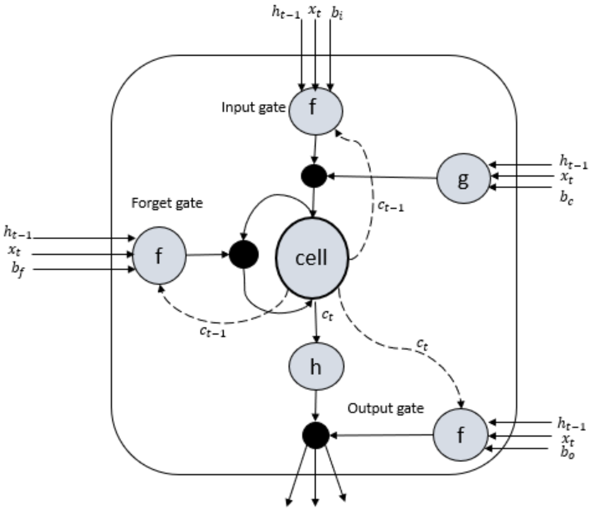

Fig. 1. The architecture of an LSTM unit [23, 24]
Inputs: x_t: Input vector, c_{t-1}: memory from previous block, h_{t-1}: output of previous block, b: Bias Outputs: h_t: the output of current block, c_t: memory from the current block

Access to both past (left) and future frames is essential for sequences labelling tasks. However, the LSTM's hidden state h_t takes information only from the past frame, without having information from the future frame. Bidirectional LSTM (BiLSTM) [25] as an elegant solution presents each sequence forwards and backwards as two separate hidden states to capture past and future information, respectively.

2.2 Existing automate pain detection models from facial expressions

Various deep learning (*i.e.*, multiple hidden layer) and non-deep learning (*i.e.*, single hidden layer) approaches have been proposed to detect pain from facial expressions, with significant progress made in this research area recently. In terms of classification problems, some traditional non-deep learning algorithms such as Support Vector Machine (SVM) have been used for the classification of features in facial expressions. In terms of non-deep learning feature extraction techniques, Active Appearance Models (AAM) and Active Shape Model (ASM), Gabor wavelets, Local Binary Pattern (LBP) were applied to extract features for this task. For example, [26, 27] used AAM based features combined with SVM classifiers for pain detection. [28, 29] applied Gabor wavelets as the main components of their filter banks and LBP features with SVMs, respectively.

Recently, with major progress in deep learning abilities and increasingly available large training data to work with, deep learning algorithms, which have a good ability to reveal intrinsically concealed patterns in complex datasets (*e.g.*, images), have been applied in feature extraction and classification problems. Deep models such as convolutional networks and deep belief and are recognized to improve feature extracting process [10, 30]. For example, significantly accurate results were achieved in pain detection from facial expression by using a pre-trained CNN for features extraction in the UNBC-McMaster Shoulder Pain Archive database [17]. Furthermore, [22] proposed a real-time regression framework based on the RNN to estimate pain levels from facial expressions by extracting features from pre-trained CNN and combining them with RNN as a new model. Using the same technique, [31, 32] extracted facial features from pre-trained VGGFaces, and then integrated them into a LSTM to utilize the temporal relationships between video frames. In a new and different painful facial expression database MIntPAIN [2, 16], a pre-trained CNN (VGGFace) and LSTM were applied in a fusion algorithm for spatial-temporal analysis considering Depth (D), and Thermal (T) accompanied by chromatic (RGB) video data to detect pain in five classes. In [33], a three stream network with three different feature extraction techniques including the appearance Histogram of Oriented Gradients (HOG), CNN, and the shape features using handcrafted algorithms and the Relevance Vector Machines (RVM) used to estimate the pain. In [34], proposed an automated pain detection system including two machine learning systems: an Automated Facial Expression Recognition (AFER) system that computes the frame-level confidence scores for single AUs and a Multiple Instant Learning (MIL) system that performs the sequence-level pain prediction based on contributions from a pain-relevant set of AU combinations. More details about the automatic pain recognition approaches are explained in a survey paper published recently [35].

2.3 Ensemble neural networks

An ensemble model, following notion of '*The Wisdom of Crowds*', can be described as a composition of multiple weak learners to form one single learner with expected higher predictive performance. The weak learner is defined as a learner that performs slightly better than random guessing [36]. Ensembles of learning algorithms have been effectively used in many computer vision problems to improve the classification performance [15, 37]. According to [14] ensemble learning is effective method since: "1) the training phase does not provide enough data to shape a single finest classifier; 2) an ensemble using separate starting points could better estimated the finest result; 3) an ensemble may expand space for a better approximation". Also, ensemble learning algorithms improve the generalization ability. Ding and Tao (2017), used ensemble CNN for video-based face recognition [38]. Their model outperforms previous approaches such as Deep Face [18], DeepID2+ [19], and VGG Face [39]. According to [40], , a neural network ensemble can be designed by altering the initial weights, the network architecture, and the training set. The combined decision created by the ensemble method is less expected error than the decision produced by other individual networks [41]. A Horizontal and Vertical Ensemble methods proposed to enhance the classification performance of deep neural networks. Based on their results both linear Horizontal Voting and Horizontal Stacked Ensemble methods can strongly enhance the performance of deep learning classification [42].

3. The proposed ensemble deep learning framework

The novelty of this study is to propose a new ensemble deep learning model (EDLM) to classify pain intensity in multi-levels (five classes) from facial expression video frames data. The block diagram of the proposed system is shown in Fig. 2. The pre-processing and normalization are applied on dataset before feeding the images into the proposed deep learning model. The EDLM consists of two sections including the early fusion and late fusion. In the early fusion, a combination of pre-trained CNN and linear PCA is used to extract and select features. Then, the extracted features are transferred in the late fusion for classification. An ensemble three stream CNN+RNN hybrid deep learning network is used in late fusion to classify pain levels in five classes. In the following, the details of the early fusion, late fusion, and entire EDLM algorithm is explained.

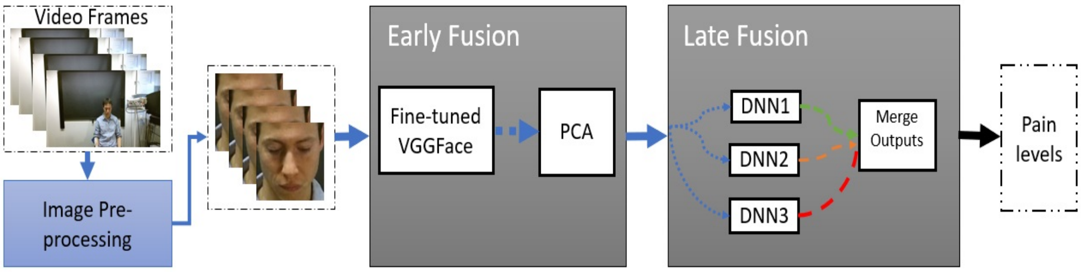

Fig. 2. Block diagram of the proposed ensemble deep learning model (EDLM) to detect pain in multi-classes from facial expressions.

3.1 Early fusion

To design the proposed EDLM model, the first step is to design early fusion feature extraction section. In addition, the pre-processed data is transferred in the early fusion algorithm to extract features. The early fusion section contains of the fine-tuned VGGFace pre-trained with Faces [39] which its outputs combined with PCA (see Fig. 3).

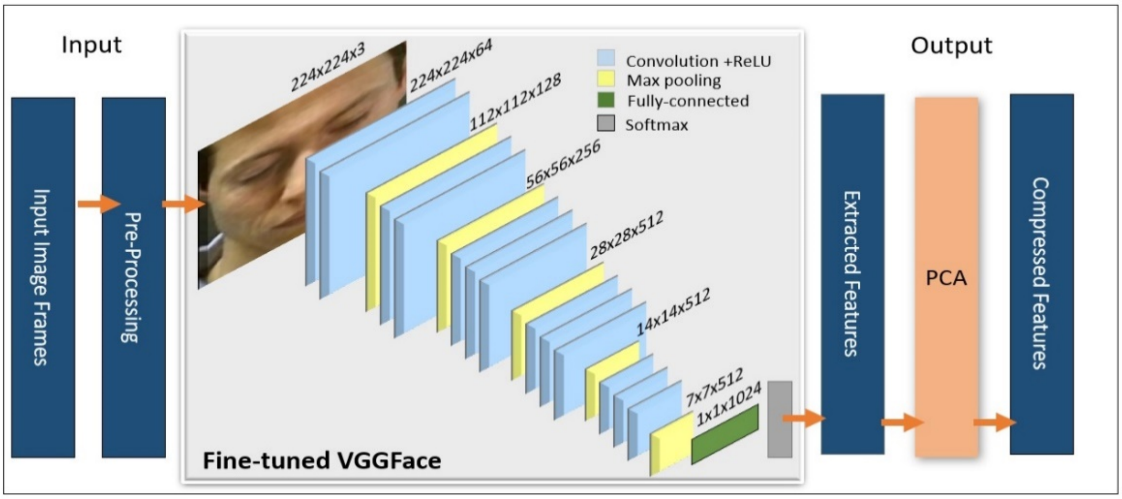

Fig. 3. Early fusion step of the EDLM for feature extraction and selection by integration fine-tuned VGGFace and PCA

In the computer vision field, transfer learning is usually expressed using pre-trained models such as a model trained on a large benchmark dataset to solve a problem. Several of the state-of-the-art techniques used transfer learning solution to generate results in image classification [10]. The VGGFace has five convolution blocks and three fully connected layers (fc6, fc7, and fc8). For fine-tuning it, a Dense connected model at top of the VGGFace model is created and the convolution layers are freeze, then data normally fed to the network [43]. Convolution neural networks-based methods can derive deep feature extraction from a set of training images. However, one challenge in this task is that the dimension of the extracted image features can increase dramatically with the addition of more network layers [44]. To resolve this problem, after using deep learning to extract image features, the PCA algorithm is used in this study to achieve dimension reduction. The study adopts PCA, as it is a dimensionality reduction method that is useful for diverse applications (*e.g.*, image compression, facial feature extraction, face recognition and finding the patterns from large dimensional images) [45, 46]. This method can also help us choose the best set of data dimensions that will make the model perform better, and to increase efficiency of the algorithm performance [47]. There is a total of 125280 features, which have been extracted from the training data set, calculated according to the input shape of the extracted features. For the training data set, these are denoted as (31320, 4) where the number 34800 refers to the number of training images and so, we are able to obtain a product 31320 × 4 = 125280. In addition, the 4 distinct output features (per image) extracted from the fine-tuned VGG-Face are transferred into the PCA algorithm with an aim to reduce the dimensionality of the extracted features and also to increase efficiency of the classification algorithm. It would thus be of interest to be able to discover "sparse principal components" such as sparse vectors spanning a low-dimensional space. To achieve this, it is necessary to reduce some of the explained variance and the orthogonality of the principal components. The explained variance for each component is calculated by Python software.

The dimensionality reduction process is achieved through an orthogonal, linear projection operation. Without loss of generality, the PCA operation can be defined as [48]:

$$Y = XC \qquad (6)$$

with

$$Y \in R^{S \times P}$$

is the projected data matrix that contains *P* principal components of *X* with,

$$P \leq N.$$

So, the key is to find the projection matrix

$$C \in R^{N \times P}$$

which is equivalent to find the eigenvectors of the covariance matrix of X, or alternatively solve a singular value decomposition (SVD) problem for X.

$$X = \mathrm{U}\Sigma\mathrm{V}^{T} \qquad (7)$$

where

$$U \in R^{s \times s}$$

and

$$V \ \in R^{N \times N}$$

are the orthogonal matrices for the column and row spaces of *X*, and Σ is a diagonal matrix containing the singular values,

$$\lambda_n, \qquad \text{for n} = 0, \cdots, \text{N}-1$$

non-increasingly lying along the diagonal. It can be shown that the projection matrix C can be obtained from the first *P* columns of *V* with

$$V = [v_1, \ldots, v_N] \qquad (8)$$

and

$$C = [c_1, \ldots, c_P] \qquad (9)$$

where

$$v_n \in \ R^{N \times 1}$$

is the n^{th} right singular vector of X, and

$$c_n = v_n.$$

In fact, the singular values contained in Σ are the standard deviations of X along the principal directions in the space spanned by the columns of C. Therefore, λ_n^2 becomes the variance of X projection along the n^{th} principal component direction. It is believed that variance can be explained as a measurement of how much information a component contributes to the data representation. One way to examine this is to look at the cumulative explained variance ratio of the principal components, given as [48]:

$$R_{cev} = \frac{\sum_{n=1}^{P} \lambda_n^2}{\sum_{n=1}^{N} \lambda_n^2} \tag{10}$$

Fig. 4 describes that selecting 3 components can preserve majority of the total variance of the input data. A vital part of using PCA in practice is the ability to estimate how many components are needed to describe the data. This can be determined by looking at the cumulative *explained variance ratio* as a function of the number of components. This graph quantifies how much of the total, 4-dimensional variance is contained within the components. For example, we see that with the first 1 component contain approximately 48% of the variance, while we need around 3 components to describe close to 100% of the variance.

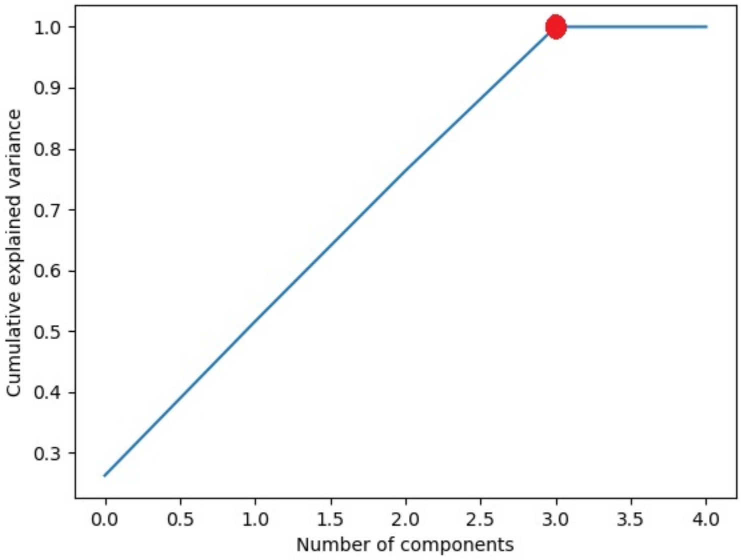

Fig. 4. Number of components to select from extracted features by PCA

3.2 Late fusion section

In the late fusion part of the proposed EDLM used as the classification section, an ensemble deep learning network is designed in varying initial weights and network architecture. As discussed in the *Related Works* section, ensemble learning is an effective method and can improve the generalization ability of classification. Since the data is video and contains video image frames, and RNNs suited for sequential data we used temporal information to feed into RNNs. The training of RNNs act as back-propagation algorithm [25].

The proposed algorithm was tested in a different version. The experimental results indicated that using hybrid CNN+RNN in late fusion has more accurate results than networks that only include RNN in late fusion. Therefore, three independent and hybrid CNN+RNN deep learning methods are designed and their outputs are merged. The merged output used to classify pain intensity. These three independent and hybrid deep learning networks are DNN1, DNN2, and DNN3 which are developed using different parameter, weight, and architecture. The configurations of these networks are described in Table 1. As can be seen from Table 1, DNN1 and DNN2 contain two CNNs with Conv2D architecture which their output shift in stack way to a BiLSTM. However, DNN1 and DNN2 are different in weighting. For DNN3, a different architecture of CNN+RNN is used. In addition, a CNN with Conv1D is selected and its output is transferred into a LSTM. Fig. 5 illustrates the late fusion architecture of the proposed EDLM model.

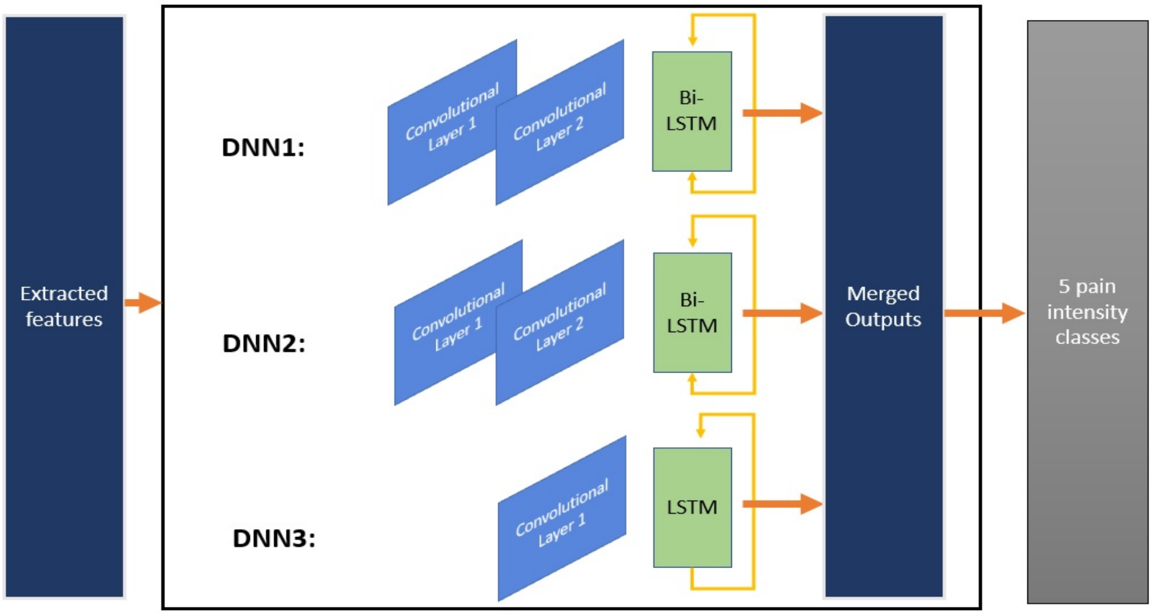

Fig. 5. Late fusion step of the EDLM based on ensemble deep neural network.

Table 1. Properties of DNN1, DNN2, and DNN3 proposed in the late fusion stage.

DNN	Convolution layer 1	Convolution layer 2	RNN

DNN1	type = conv2d,
	filter number = 256,
	activation = ReLU,
	input shape = (1,5)
		type = conv2d
		filter number = 256,
		activation = ReLU,
		input shape = (1,5)
			type = BiLSTM,
			filter number = 256,
			dense = 4096,
			drop out = 0.5,
			activation = ReLU

DNN2	type = conv2d,
	filter number = 128,
	activation = ReLU,
	input shape = (1,5)
		type = conv2d
		filter number = 128,
		activation = ReLU,
		input shape = (1,5)
			type = BiLSTM,
			filter number = 32,
			dense = 4096,
			drop out = 0.5,
			activation = ReLU

DNN3	type = conv1d,
	filter number = 256,
	activation = ReLU,
	input shape = (1,5)
		None	type = BiLSTM,
			filter number = 128,
			dense = 4096,
			drop out = 0.5,
			activation = ReLU

3.3 The EDLM algorithm design

The details of the proposed EDLM method are summarized in Algorithm 1. During experimentation optimization for the early fusion feature extraction section, the model ran by 50 epoch and 48 batches. However, in the late fusion, the model performed by 5 epoch and 48 batches. To estimate the skill of the algorithm, the cross-validation method involved by repeating 10 times.

Algorithm 1: The proposed EDLM algorithm

```
Procedure EDLM (input, n, j, batch)
        Pre-process (input)
        for k ← 0, n do
                finetune (VGG-Face)
                for epoch ← 0, j do
                        features ← train (finetune (VGG-Face))
                end for
                SF ← PCA (features)
                GN ← Calculate (GN)
                for epoch ← 0, j do
                        o1 ← DNN1(SF)
                        o2 ← DNN2(SF)
                        o3 ← DNN3(SF)
```

```
            out ← merge (o1, o2, o3)
            out ← GN (48)
            train (model (SF, out))
        end for
    end for
end procedure
```

4. Experimental configuration and databases

In this study, the objective algorithm (EDLM) and all the other comparative algorithms are built under an Intel core *i7* @ 3.3 GHz and 16 GB memory computer. *Python software* [49] was used for the model construction and prototyping, since it has freely available libraries suits for deep learning such as *Keras* [50], *TensorFlow* [51], *Scikit-learn* [52], *Matplotlib* [53]. *Keras* allows for easy and fast prototyping and supports both convolutional networks and recurrent networks. *Matplotlib* as a *Python* 2D plotting library is used for plotting and statistical analysis of modelling data. The selected database and evaluation metrics are explained as following.

4.1 The MIntPAIN database

To establish the robustness of the proposed EDLM model, we used two databases includes the MIntPAIN database [2, 16] and the UNBC-McMaster Shoulder Pain dataset [17]. The MIntPAIN database includes pain video data taken by electrical stimulation in five levels (Level 0 no pain, and to Level 4 is the highest level) to 20 subjects. Each subject includes two trials, and each trial includes 40 sweeps of pain stimulation. In this research work, a dataset of all RGB images from 20 subjects is selected. The number of no pain video sequences are more than others. Therefore, based on the specific character of the database it is likely that any model gets biased towards the prediction of no-pain at the cost of missing pain frames. Using imbalance data is basically intentionally biasing data to get an interesting result. To deal with this issue, in this study the database was balanced using under resampling techniques to reduce the majority class (no-pain class). So, some no pain sequences have been removed.

The resampling technique was applied on the selected dataset since a few subjects were missing for some sweeps and there was not an equal proportion for each class as well. Therefore, the under-sample technique was applied to reduce the majority class, and some no painful sequences (Lable0) were removed. The total of 34800 video frames is selected for experimentation in this research. Fig. 6 shows the samples of the selected dataset.

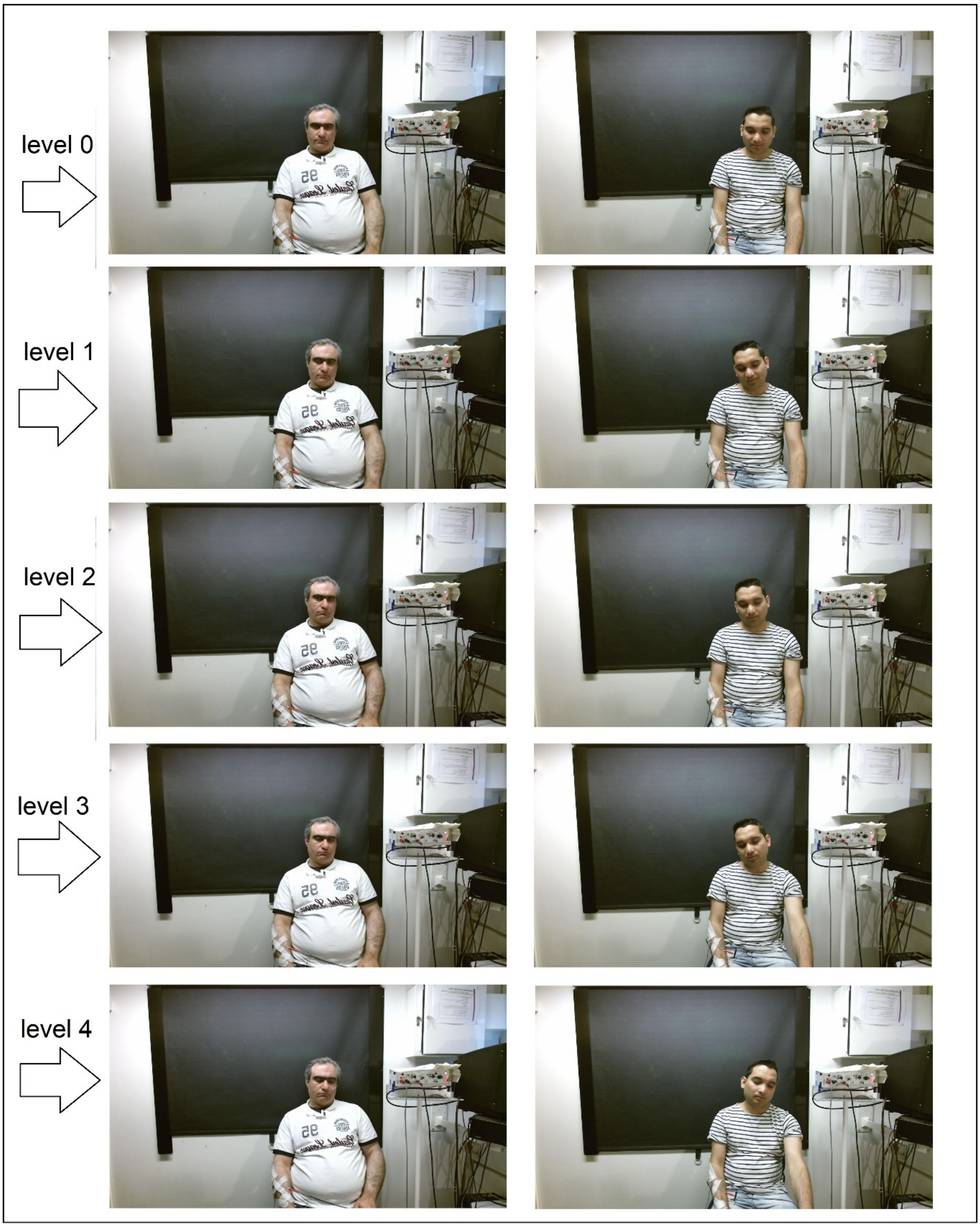

Fig. 6. Samples of selected dataset of MIntPAIN database [2, 16].

The selected dataset was pre-processed by removing noises and backgrounds from each video frames. The pre-processing includes face detecting, cropping, and centralizing applied on the video frames. Then, the images were normalized before feeding images to the proposed model. Moreover, the OpenCV face recognition algorithm was used to detect faces from noisy pictures. Then, face detected images were cropped and centralized (see Fig. 7). Finally, the pre-processed data was reshaped to

224×224×3 dimensions to transfer into VGGFace pre-trainer. To normalize the pixel values for both train and test datasets, the data was rescaled to the range [0,1]. This includes converting the data type from integer to floats and splitting the pixel values by the highest value [54].

$$Normalize{:}R\rightarrow R{:}x\rightarrow\frac{x}{d} \qquad d=\max_{x\,\in\, image}\parallel x\parallel \qquad (6)$$

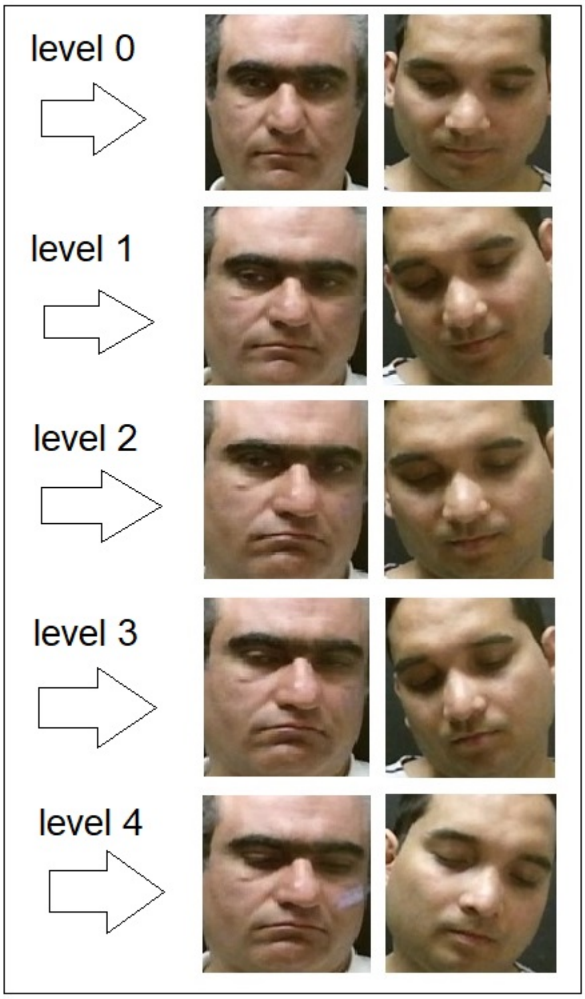

Fig. 7. Examples of video frames per 5 level after removing backgrounds, cropping, and resizing.

4.2 The UNBC-McMaster Shoulder Pain database

To prove the generality of the proposed EDLM model, the experiment was conducted on the UNBC-McMaster Shoulder Pain dataset [17] and competitive results were obtained. The database provides the image's frames of video sequences from patients suffering shoulder pain. Each frame of the database was coded in terms of PSPI score among 0 to 15 scales. The database provides 200 sequences across 25 subjects, which totals 48,398 image frames. The number of no pain images PSPI score labels are higher than the other labels and the number of images with the PSPI labels greater than 6 are only a few within this database. Therefore, based on the specific character of the database it is likely that any model can be biased towards the prediction of no-pain at the cost of missing pain frames. Using imbalance data is basically intentionally biasing data to get an interesting result. To deal with this issue, in this study the database was balanced using under resampling techniques to reduce the majority class (no-pain class). We balanced the database is by under-resampling technique to reduce the majority class (no-pain class) and 10,783 images were thus employed in this research. For classifying pain into five levels, the database was divided into five parts including (PSPI = 0), (PSPI = 1), (PSPI = 2 and 3), (PSPI = 4 and 5) and (PSPI > = 6). Fig. 8 shows samples of the UNMC-McMaster Shoulder Pain database for some classes.

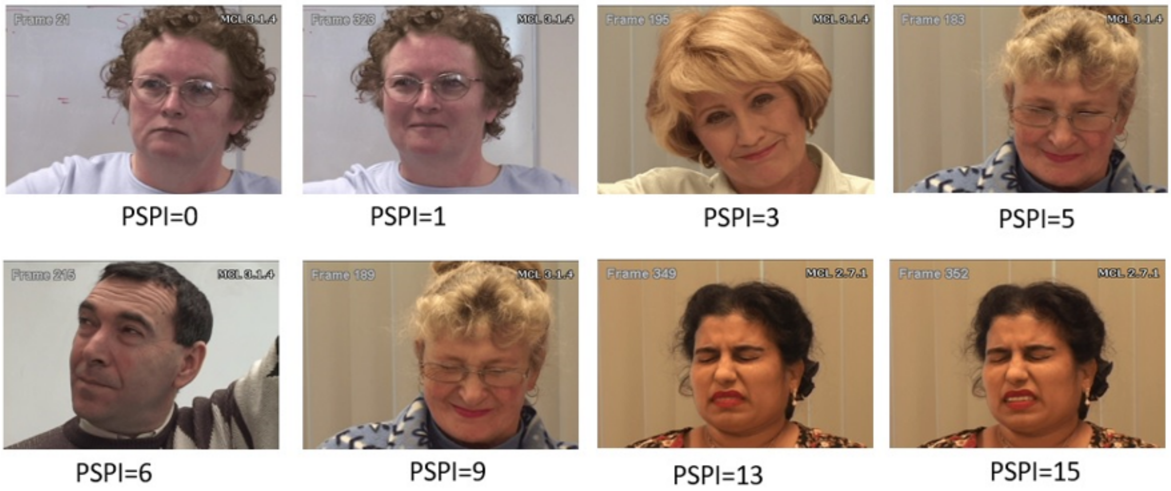

Fig. 8 Image frame samples of the UNBC-McMaster Shoulder Pain Achieve database [17].

4.2 The evaluation metrics

To train, test, and evaluate the proposed EDLM ensemble model, this section provides several empirical results of the modelling experiments carried out and evaluations in comparison with other models developed during experimentation and previous researches using MIntPAIN database. To enable rigorous evaluations of the proposed EDLM model in respect to the counterpart models, several performance evaluations measures, including the Classification Mean Absolute Error (MAE), Mean Squared Error (MSE), Accuracy, AUC and F-score were utilized. Mathematically, the metrics are stated as follows where:

$e = e_{experimental} - e_{true}$ and N = number of errors:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|e_i| \tag{11}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 \tag{12}$$

We used some metrics such as accuracy f-measure to measure performance of the algorithm. The mathematical formula of them is as following.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

$$F = 2 \times \frac{Recall \times Precision}{Recall + Precision} \tag{16}$$

$$TPR = \frac{TP}{(FN + TP)} \tag{17}$$

$$FPR = \frac{FP}{(TN + FP)} \tag{18}$$

Where True Positive (TP) is the cases are predicted YES, and the actual output is YES and True Negatives (TN) is the cases are predicted NO, and the actual output is NO. False Positives (FP) is the cases are predicted YES, and the actual output is NO. False Negatives (FN) is the cases are predicted NO, and the actual output is YES [55]. True Positive Rate (TPR) corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all positive data points. False Positive Rate (FPR) corresponds to the proportion of negative data points that are mistakenly considered as positive, with respect to all negative data points.

5. Results and discussions

In this section we train and test our proposed framework in two different databases includes MIntPain and UNBC-McMaster Shoulder Pain databases. Next, the evaluated results compared with the baseline model and the state-of-the-art researches.

5.1 The MIntPAIN database results

The features have been extracted and selected by early fusion finetuned VGGFace and PCA. The early fusion algorithm to reach its best performance used 50 epochs. Fig. 9 illustrates the accuracy and the loss error encountered in the early fusion in the EDLM model. This figure shows the average number of the accuracy for 10 cross validation during 50 epochs. As it is indicated in Fig.9 the accuracy level has been reached to the its highest level by 81% in epoch = 50. It has been started from 32% in epoch 1 and gradually has been increased. The red line in this figure shows the loss value average for 10 Cross validation and shows a decreasing amount in loss level by increasing epoch. The loss has been reached in the lowest level by 0.18 in epoch 50.

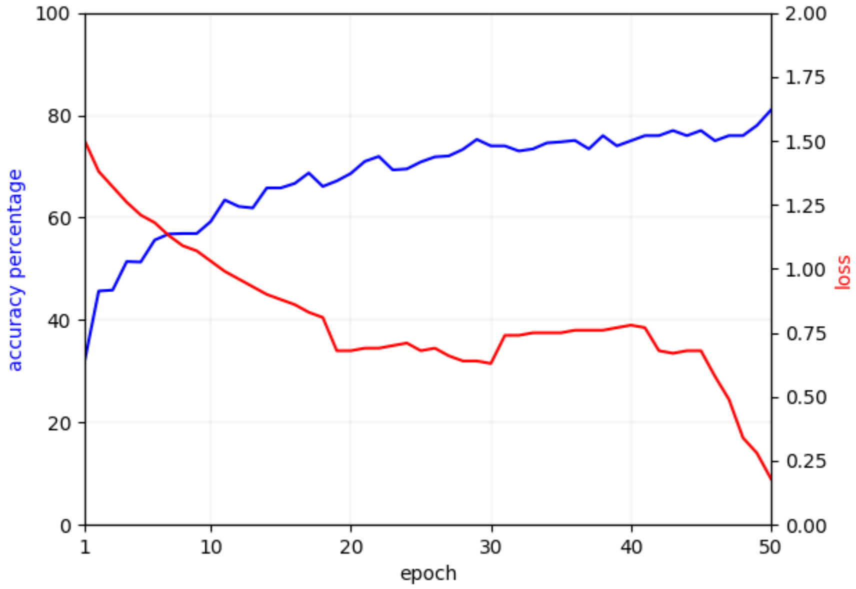

Fig. 9. Accuracy and loss error during 50 epochs in the early fusion of the EDLM model in the MIntPAIN database.

Later, the proposed classifier which is late fusion here has been trained and tested by selected features. Fig. 10 shows the accuracy and loss level during 5 epochs in average of 10 cross validation for late fusion. At first, accuracy has been started by 81% and then from the second epoch it reaches to 92.26% in epoch 5. The red graph in Fig. 10 shows the MSE level in average. As it is shown in this graph in epoch one the MSE equal to 0.06 but by repeating testing and training in epoch 5 it reached to its lowest level by 0.028.

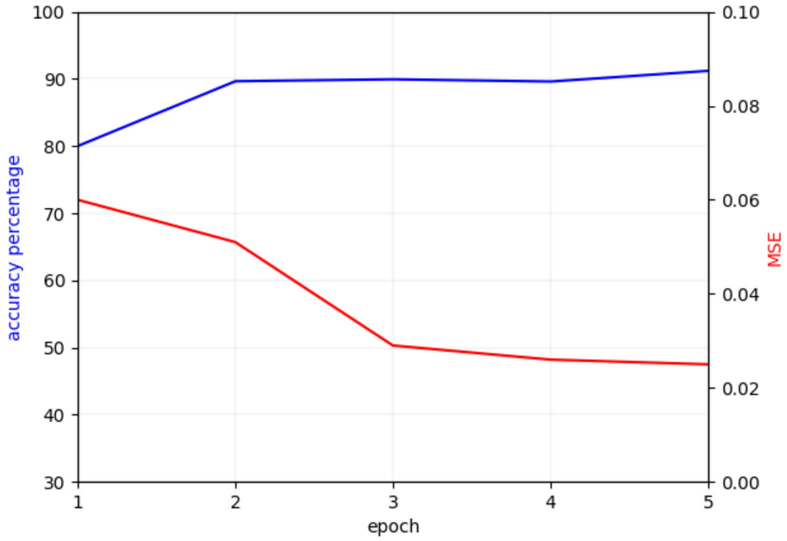

Fig. 10. Accuracy and MSE during 5 epochs in the late fusion of the EDLM model in the MIntPAIN database.

Table 2 and Fig. 11 indicate the obtained results of the proposed EDLM on the MIntPAIN database measured by accuracy, AUC, MAE, and MSE based on 10-fold cross validation.

Table 2. The average performance, best result, and worst results of the proposed model (EDLM) on MIntPAIN database for 10-fold cross validation.

Results	MSE	MAE	Accuracy	AUC
Average	0.0245	0.0341	92.26%	93.67%
Best	0.02102	0.028	95%	95.2%
Worst	0.03056	0.039	89%	91.4%

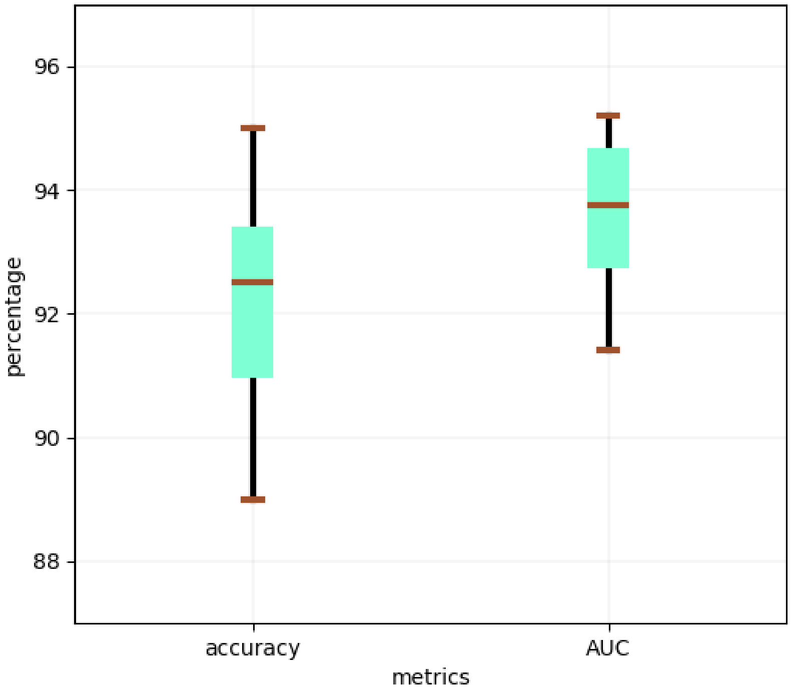

Fig. 11. Box plots of Accuracy and AUC for the proposed EDLM model in the MIntPAIN database.

Fig. 11 displays the accuracy and AUC of the proposed EDLM model in the box plot. It shows the distribution of data based on minimum, first quartile, median, third quartile, and maximum. Median is shown as yellow, minimum and maximum shown as blue lines. Median is demonstrated by the middle value of the accuracy and AUC. The first quartile shows the middle number between the smallest number and the median of the dataset. Third quartile shows the middle value between the median and the highest value of the dataset.

Other popular evaluation metrics such as f-score and precision also have been exploited to evaluate the performance of the proposed EMDL model and the results show optimum and effective ranges of effectiveness per each class. The performance of the proposed EDLM model shows a significant correctness per five classes measured by AUC ROC (Receiver Operating Characteristics) curve metric.

Table 3 indicates the accuracy, AUC, f-score, and precision for each class with no-pain, pain level 1, pain level 2, pain level 3, and pain level 4.

Table 3. Average pain level per five classes based on accuracy, f-score, precision, AUC metrics in the MIntPAIN database.

Metrics	**No pain**	**Pain 1**	**Pain 2**	**Pain 3**	**Pain 4**
AUC	87.3%	84%	85%	89%	91%
Precision	85.2%	85%	83%	88%	88%
f-score	86%	82%	82.2%	86.2%	90%
Accuracy	92.4%	89%	88%	93%	92%

The accuracy of the proposed EDLM model is assessed by TPR and FPR analysis and results show effectiveness of it by obtaining higher values for TPR and lower values for FPR in five classes.

5.2 The UNBC-McMaster Shoulder Pain database results

To prove the generality of the proposed EDLM model, the experiment was conducted on the UNBC-McMaster Shoulder Pain dataset and the obtained results indicate that the proposed EDLM framework has high performance in this database. In this database we used PSPI labels per each frame. To enable rigorous evaluations of the proposed EDLM model in respect to the counterpart models, several performance evaluations measures, including the MAE, MSE, Accuracy, and AUC were utilized. Table 4 indicates the obtained results of the proposed EDLM on the UNBC-McMaster Shoulder Pain database measured by accuracy, AUC, MAE, and MSE based on 10-fold cross validation.

Table 4. The average performance of the proposed model (EDLM) in the UNBC-McMaster Shoulder Pain database for 10-fold cross validation.

MSE	**MAE**	**Accuracy**	**AUC**
0.081	0.103	86%	90.5%

5.3 Discussion

We compared the obtained results from the EDLM with a baseline model which is designed based on a standard VGG-Face and one stream LSTM model. Table 5 shows the comparison results obtained by the EDLM proposed framework with the baseline model results. As it is indicated in this table the proposed EDLM has higher performance than the standard baseline model.

Table 5. The comparison of the obtained AUC and accuracy from the EDLM and the baseline model in the MIntPAIN database.

Classification models	**AUC**	**Accuracy**
VGGFace + 1 stream LSTM	87%	83.4%
The proposed EDLM model	*93.67%*	*92.26%*

The time complexity of the proposed EDLM algorithm has also been measured in two databases and compared with two other baseline models which have been developed during experimental. Table 6 shows the learning time of the EDLM for two databases in comparison with two different baseline models. As is indicated in Table 6, the total time complexity of the proposed EDLM algorithm for the UNBC-McMaster Shoulder Pain database is 5900 s and the time complexity of it for the MIntPAIN database is 41700 s.

As a result, the most time-consuming section of the EDLM is feature extraction section and adding more streams in the classifier has not affected the algorithm speeds and efficiency. On the other hand, the selected database and the required number of epochs are important factors which affect the complexity and learning time of the algorithm.

Table 6. The time complexity of the proposed EDLM in compare with other baseline algorithm in the UNBC-McMaster Shoulder Pain database and MIntPAIN database.

Models	**Database**	**Early fusion Time complexity (based on second) and number of applied epochs**	**Late fusion Time complexity (based on second) and number of applied epochs**	**Sum of the Time complexity**
VGGFace + 1 stream LSTM	UNBC-McMaster	10400 / 5	560 / 5	10960
VGGFace + 1 stream LSTM	MIntPAIN	108000 / 50	1600 / 5	109600
VGGFace + PCA + 1 stream LSTM	UNBC-McMaster	5300/ 5	560 / 5	5860
VGGFace + PCA + 1 stream LSTM	MIntPAIN	40000 / 50	1600 / 5	41600
Proposed EDLM (VGGFace + PCA + 3 stream CNN-BiLSTM)	*UNBC-McMaster*	*5300 / 5*	*600 / 5*	*5900*
Proposed EDLM (VGGFace + PCA + 3 stream CNN-BiLSTM)	*MIntPAIN*	*40000 / 50*	*1700 / 5*	*41700*

The EDLM model demonstrated the highest performance in comparison with the other models and the state-of-the-art results. Table 7 indicates a comparison of the proposed EDLM method scores against other state-of-the-art procedures in pain intensity recognition. In this table the obtained results trained and tested in the both databases compared with the other research works.

Table 7. Comparing the proposed EDLM with the other state-of-the-art procedures in pain intensity recognition.

Ref	**Pain**	**AUC**	**Classifier**	**Accuracy**	**MSE**	**Database**	**Data size**
	Level	**(%)**		**(%)**			
[17]	2	83.9	SVM	-	-	UNBC-McMaster	All
[27]	2	84.7	SVM	-	-	UNBC-McMaster	All
[31]	2	93.3	CNN-LSTM	83.1	0.74	UNBC-McMaster	Down-up
[16]	3	-	CNN-RNN	61.9	-	UNBC-McMaster	Down-up
[22]	2	-	-	-	1.54	UNBC-McMaster	16657 images
[2]	5	-	CNN-LSTM	32.40	-	MIntPAIN	All
[56]	4	98.4	PCA-CNN-RNN	91.2	0.04	UNBC-McMaster	Down-up
Proposed EDLM	*5*	*93.67*	*Ensemble CNN-RNN*	*92.26*	*0.0245*	*MIntPAIN*	*34800 images*
Proposed EDLM	*5*	*90.5*	*Ensemble CNN-RNN*	*86*	*0.081*	*UNBC-McMaster*	*10783 images*

By analysing the results and comparing them with the state-of-the-art results, we can conclude as follows:

1. The proposed new feature extraction model composing fine-tuned VGGFace pre-trained and PCA significantly increased the performance of the algorithm feature extraction in compare with the standard VGG-Face.

2. The proposed ensemble deep learning model (EDLM) which integrated three independent CNN-RNN deep learners with vary in weights and structures has high performance in comparison with the baseline VGG-Face and one stream LSTM model.

3. An evaluation of the proposed model through statistical metrics and investigative plots expose that the ensemble EDLM model generates improved classification compared to the other benchmarked models in multi classes.

4. The proposed EDLM model is the optimum deep learning method resulting in a low qualified error compared with the other target models in this task.

Although the obtained results from evaluation of the newly developed EDLM model confirm its effectiveness, the feature work can use different frameworks for pain recognition such as the technique introduced in [8] which firstly recognizes the general facial expression, then if it detects pain, then use the authors' proposal to provide fine-grained pain level classification. Deep metric learning methods may also be used to achieve better performance such as Siamese networks [9]. Future work, may also consider loss functions method that perform well on imbalanced datasets [57-61].

There are some limitations in terms of the number of pain datasets from facial expressions in pain detection research. One of the challenges is that most of the research into facial expressions, especially in the area of facial pain detection, currently lacks a standard database. This makes it relatively difficult to train an accurate facial image recognition system that can act as a robust platform for recognizing the pain and modelling the subsequent pain intensity relative to any given facial image.

6. Conclusions and future work

This study was designed to support ongoing efforts in developing artificial intelligence technologies for pain detection using facial expression images, and as such, the work has proposed a newly designed, classification model with an ensemble deep learning approach. The resulting EDLM model therefore integrates the three-stream independent CNN-RNN based networks that are seen to vary in their structure and weights denoting features extracted from facial images. The proposed EDLM model then applied the fine-tuned VGGFace algorithm, integrated with the PCA approach to extract features from facial images. Finally, the ensemble deep learning model that includes three independent CNN-RNN was designed and tested for its classification accuracy.

The proposed EDLM model has been evaluated comprehensively through the MIntPAIN and UNBC-McMaster Shoulder Pain datasets. The evaluated results indicate that the proposed ensemble deep learning model has an improved performance relative to the conventional method such as a single hybrid deep learning model adopted for this task. The extensive evaluation of the EDLM model, through statistical metrics and diagnostic plots, reveals its capability to generate superior classification of facial images and its features compared with the other benchmarked models. Therefore, the deep learning EDLM model is found to attain an optimal accuracy evidenced by a relatively lower error compared with the other benchmarked models. The promising capabilities of the deep learning EDLM model indicates that a future study may advance this algorithm in different types of pain face images and video frame databases to further accelerate the efficiency and effectiveness of feature extracting of images for more broader real-time applications in health informatics and medical diagnosis areas.

Acknowledgments

This research has been supported by an Australian Government Research Training Program Scholarship and was funded by the Australian Research Council (ARC) (grant number LP150100673) supported by Nexus eCare Pty Ltd. The authors declare no conflict of interest.

References

[1] D. L. Martinez, O. Rudovic, and R. Picard, "Personalized automatic estimation of self-reported pain intensity from facial expressions," presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, Hawaii, 2017.

[2] M. A. Haque *et al.*, "Deep multimodal pain recognition: a database and comparison of spatio-temporal visual modalities," in *2018 13th IEEE International Conference on Automatic Face &*

Gesture Recognition (FG 2018), Xian, China, 2018: IEEE, pp. 250-257, doi: 10.1109/FG.2018.00044.

[3] C. M. A. Ilyas, M. A. Haque, M. Rehm, K. Nasrollahi, and T. B. Moeslund, "Facial Expression Recognition for Traumatic Brain Injured Patients," in *VISIGRAPP (4: VISAPP)*, 2018, pp. 522-530.

[4] J. Klonovs *et al.*, *Distributed computing and monitoring technologies for older patients*. Switzerland AG.: Springer, 2016.

[5] P. Ekman and W. V. Friesen, *Facial action coding system: Investigator's guide*. Palo Alto, CA.: Consulting Psychologists Press, 1978.

[6] K. M. Prkachin and P. E. Solomon, "The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain," *Pain*, vol. 139, no. 2, pp. 267-274, 2008.

[7] M. Kunz, A. Gruber, and S. Lautenbacher, "Sex differences in facial encoding of pain," *The Journal of Pain*, vol. 7, no. 12, pp. 915-928, 2006.

[8] C. Zhang, P. Wang, K. Chen, and J.-K. Kämäräinen, "Identity-aware convolutional neural networks for facial expression recognition," *Journal of Systems engineering and Electronics*, vol. 28, no. 4, pp. 784-792, 2017.

[9] X. Liu, B. Vijaya Kumar, J. You, and P. Jia, "Adaptive deep metric learning for identity-aware facial expression recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 20-29.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, Nevada, USA, 2012, pp. 1097-1105.

[11] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Columbus, Ohio, 2014, pp. 152-159.

[12] G. Tsoumakas and I. Vlahavas, "Random k-labelsets: An ensemble method for multilabel classification," in *European conference on machine learning*, Warsaw, Poland 2007: Springer, pp. 406-417.

[13] B. Liu, Q. Cui, T. Jiang, and S. Ma, "A combinational feature selection and ensemble neural network method for classification of gene expression data," *BMC bioinformatics*, vol. 5, no. 1, p. 136, 2004.

[14] T. G. Dietterich, "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*, Günzburg, Germany 2000: Springer, pp. 1-15, doi: https://doi.org/10.1007/3-540-45014-9_1.

[15] K. Simonyan and A. Zisserman, "VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION," in *ICLR*, San Diego, 2015, pp. 1-14.

[16] M. Bellantonio *et al.*, "Spatio-temporal pain recognition in cnn-based super-resolved facial images," in *Video Analytics. Face and Facial Expression Recognition and Audience Measurement*, Cancun, Mexico, 2016: Springer, pp. 151-162.

[17] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and I. Matthews, "Painful data: The UNBC-McMaster shoulder pain expression archive database," in *Face and Gesture 2011*, Santa Barbara, CA, USA 2011: IEEE, pp. 57-64.

[18] Y. Taigman, M. Yang, M. A. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Columbus, Ohio, 2014, pp. 1701-1708.

[19] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Boston, Massachusetts, 2015, pp. 2892-2900.

[20] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," *Remote Sensing*, vol. 7, no. 11, pp. 14680-14707, 2015, doi: https://doi.org/10.3390/rs71114680.

[21] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, Boston, Massachusetts, pp. 1-9.

[22] J. Zhou, X. Hong, F. Su, and G. Zhao, "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*, Honolulu, Hawaii, USA, 2016: IEEE, pp. 84-92.

[23] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85-117, 2015.

[24] F. A. Gers and E. Schmidhuber, "LSTM recurrent networks learn simple context-free and context-sensitive languages," *IEEE Transactions on Neural Networks*, vol. 12, no. 6, pp. 1333-1340, 2001, doi: 10.1109/72.963769.

[25] C. Dyer, M. Ballesteros, W. Ling, A. Matthews, and N. A. Smith, "Transition-based dependency parsing with stack long short-term memory," in *ACL 2015*, Beijing, 2015.

[26] A. B. Ashraf *et al.*, "The painful face–pain expression recognition using active appearance models," *Image and vision computing*, vol. 27, no. 12, pp. 1788-1796, 2009, doi: https://doi.org/10.1016/j.imavis.2009.05.007.

[27] P. Lucey *et al.*, "Automatically Detecting Pain in Video Through Facial Action Units," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, 3, pp. 664-674, 2011.

[28] G. Littlewort-Ford, M. S. Bartlett, and J. R. Movellan, "Are your eyes smiling? detecting genuine smiles with support vector machines and gabor wavelets," in *Proceedings of the 8th Joint Symposium on Neural Computation*, 2001, UC San Diego.

[29] C. Shan, "Learning local binary patterns for gender classification on real-world face images," *Pattern recognition letters*, vol. 33, no. 4, pp. 431-437, 2012.

[30] G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning lecture 6a overview of mini-batch gradient descent," 2012.

[31] P. Rodriguez *et al.*, "Deep pain: Exploiting long short-term memory networks for facial expression classification," *IEEE transactions on cybernetics*, no. 99, pp. 1-11, 2017.

[32] G. Bargshady, J. Soar, X. Zhou, R. C. Deo, F. Whittaker, and H. Wang, "A joint deep neural network model for pain recognition from face," in *Proceedings of the 4th IEEE International Conference on Computer and Communication Systems (ICCS 2019)*, Singapore, 2019: IEEE Press, pp. 52-56.

[33] J. Egede, M. Valstar, M. T. Torres, and D. Sharkey, "Automatic Neonatal Pain Estimation: An Acute Pain in Neonates Database," in *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2019: IEEE, pp. 1-7, doi: 10.1109/ACII.2019.8925480.

[34] Z. Chen, R. Ansari, and D. Wilkie, "Learning Pain from Action Unit Combinations: A Weakly Supervised Approach via Multiple Instance Learning," *IEEE Transactions on Affective Computing*, 2019, doi: 10.1109/TAFFC.2019.2949314.

[35] P. Werner, D. Lopez-Martinez, S. Walter, A. Al-Hamadi, S. Gruss, and R. Picard, "Automatic Recognition Methods Supporting Pain Assessment: A Survey," *IEEE Transactions on Affective Computing*, 2019.

[36] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119-139, 1997, doi: https://doi.org/10.1007/3-540-59119-2_166.

[37] R. Minetto, M. P. Segundo, and S. Sarkar, "Hydra: an ensemble of convolutional neural networks for geospatial land classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6530 - 6541, 2019.

[38] C. Ding and D. Tao, "Trunk-branch ensemble convolutional neural networks for video-based face recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 1002-1014, 2017, doi: 10.1109/TPAMI.2017.2700390.

[39] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *bmvc*, Swansea, UK, 2015, vol. 1, no. 3, p. 6.

[40] A. J. C. SHARKEY, "On combining artificial neural nets," *Connection Science*, vol. 8, no. 3-4, pp. 299-314, 1996.

[41] L. K. Hansen and P. Salamon, "Neural network ensembles," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 10, pp. 993-1001, 1990, doi: 10.1109/34.58871.

[42] J. Xie, B. Xu, and Z. Chuang, "Horizontal and vertical ensemble with deep representation for classification," in *ICML 2013*, Atlanta, 2013.

[43] A. Gulli and S. Pal, *Deep Learning with Keras*. Birmingham, UK: Packt Publishing Ltd, 2017.

[44] J. Ma and Y. Yuan, "Dimension reduction of image deep feature using PCA," *Journal of Visual Communication and Image Representation*, vol. 63, p. 102578, 2019.

[45] J. Li, B. Zhao, H. Zhang, and J. Jiao, "Face recognition system using svm classifier and feature extraction by pca and lda combination," in *2009 International Conference on Computational Intelligence and Software Engineering*, Wuhan, China, 2009: IEEE, pp. 1-4, doi: 10.1109/CISE.2009.5364125.

[46] R. C. Damale and B. V. Pathak, "Face Recognition Based Attendance System Using Machine Learning Algorithms," in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India 14-15 June 2018 2018: IEEE, pp. 414-419, doi: 10.1109/ICCONS.2018.8662938.

[47] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *the 27th International Conference on Neural Information Processing Systems* Montreal, Canada, 2014, vol. 2: ACM, pp. 1988-1996.

[48] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[49] M. F. Sanner, "Python: a programming language for software integration and development," *J Mol Graph Model*, vol. 17, no. 1, pp. 57-61, 1999.

[50] N. Ketkar, "Introduction to keras," in *Deep Learning with Python*. Berkeley, CA: Springer, 2017, pp. 97-111.

[51] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th Symposium on Operating Systems Design and Implementation* Savannah, GA, 2016, pp. 265-283.

[52] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825-2830, 2011.

[53] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Computing in science & engineering*, vol. 9, no. 3, p. 90, 2007, doi: 10.1109/MCSE.2007.55.

[54] N. Schertler, "Improving JPEG Compression with Regression Tree Fields," Master, Technische Universität Dresden, Dresden Germany, 2014. [Online]. Available: https://tu-dresden.de/ing/informatik/smt/cgv/ressourcen/dateien/lehre/ergebnisse_studentischer_arbeiten/masterarbeiten/nico_schertler_ss14/files/Thesis.pdf?lang=en

[55] D. M. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37-63, 2011.

[56] G. Bargshady, X. Zhou, R. C. Deo, J. Soar, F. Whittaker, and H. Wang, "Enhanced deep learning algorithm development to detect pain intensity from facial expression images," *Expert Systems with Applications*, vol. 149, no. 113305, 2020, doi: https://doi.org/10.1016/j.eswa.2020.113305.

[57] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980-2988.

[58] C. Zhang *et al.*, "Multi-imbalance: An open-source software for multi-class imbalance learning," *Knowledge-Based Systems*, vol. 174, pp. 137-143, 2019.

[59] J. Bi and C. Zhang, "An empirical comparison on state-of-the-art multi-class imbalance learning algorithms and a new diversified ensemble learning scheme," *Knowledge-Based Systems*, vol. 158, pp. 81-93, 2018, doi: Show more

https://doi.org/10.1016/j.knosys.2018.05.037.

[60] C. Zhang, J. Bi, and P. Soda, "Feature selection and resampling in class imbalance learning: Which comes first? An empirical study in the biological domain," in *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2017: IEEE, pp. 933-938.

[61] C. Zhang, C. Liu, X. Zhang, and G. Almpanidis, "An up-to-date comparison of state-of-the-art classification algorithms," *Expert Systems with Applications*, vol. 82, pp. 128-150, 2017.

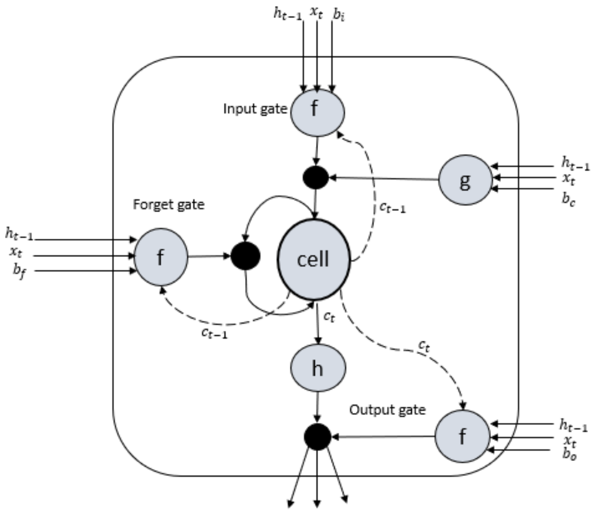

Fig. 1. The architecture of an LSTM unit [23, 24]
Inputs: x_t: Input vector, c_{t-1}: memory from previous block, h_{t-1}: output of previous block, b: Bias Outputs: h_t: the output of current block, c_t: memory from the current block

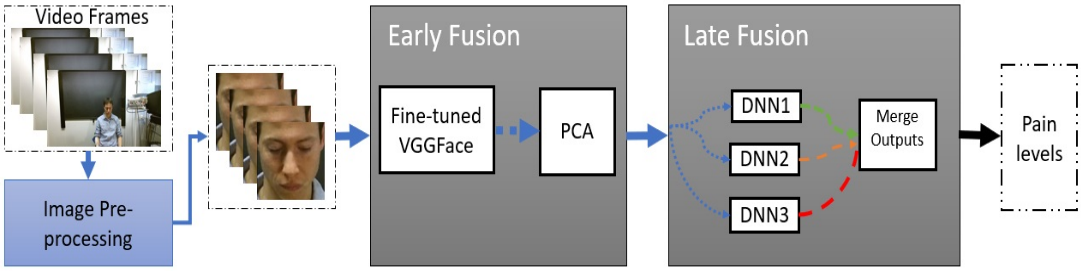

Fig. 2. Block diagram of the proposed ensemble deep learning model (EDLM) to detect pain in multi-classes from facial expressions.

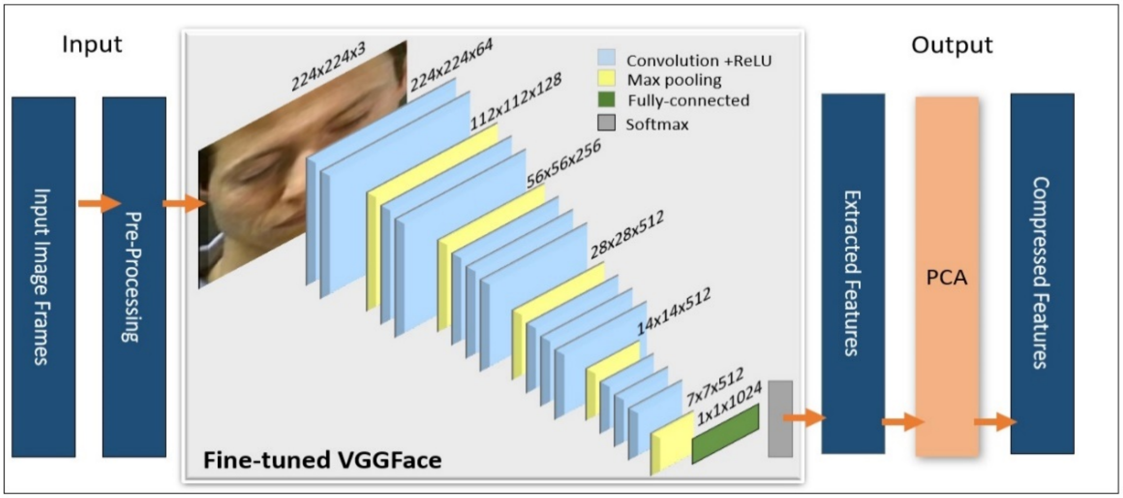

Fig. 3. Early fusion step of the EDLM for feature extraction and selection by integration fine-tuned VGGFace and PCA

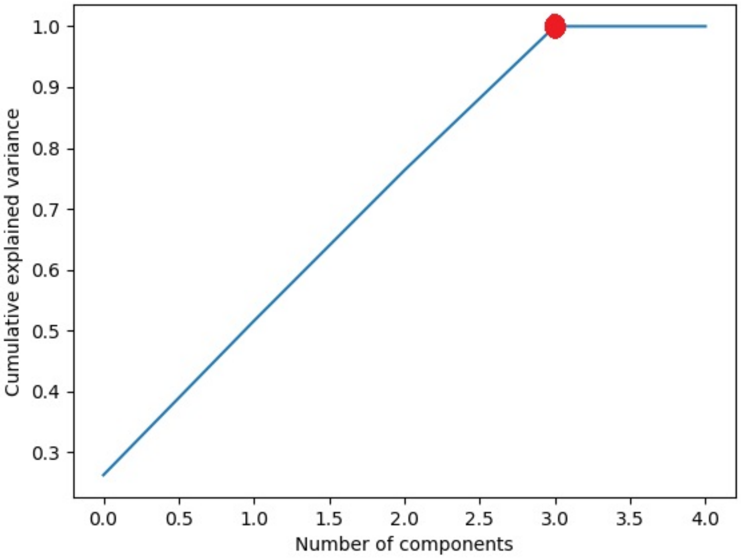

Fig. 4. Number of components to select from extracted features by PCA

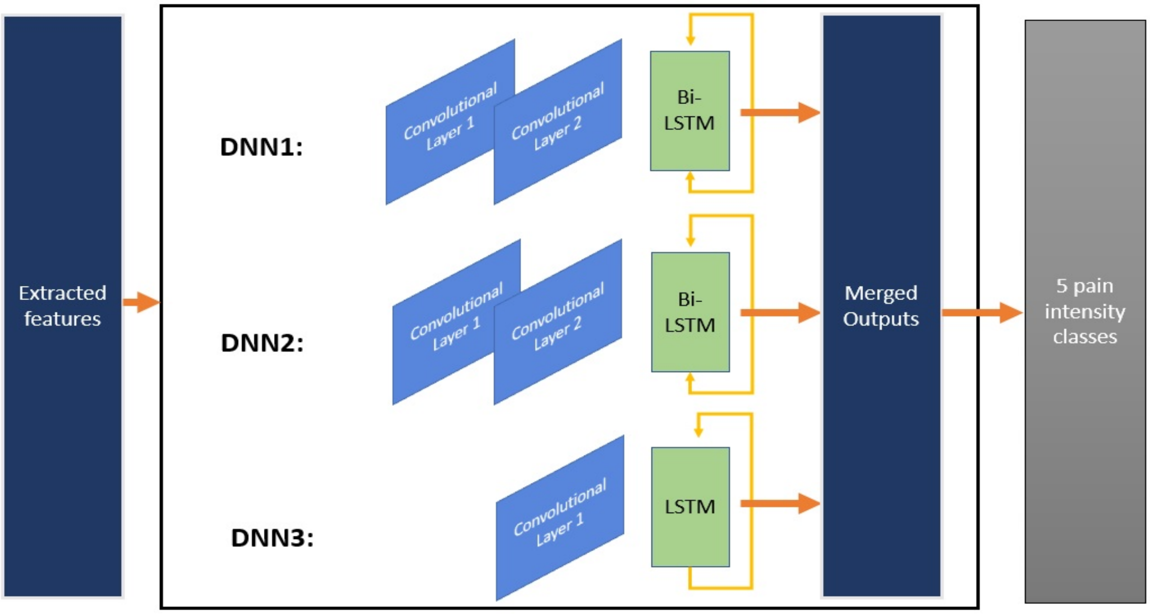

Fig. 5. Late fusion step of the EDLM based on ensemble deep neural network.

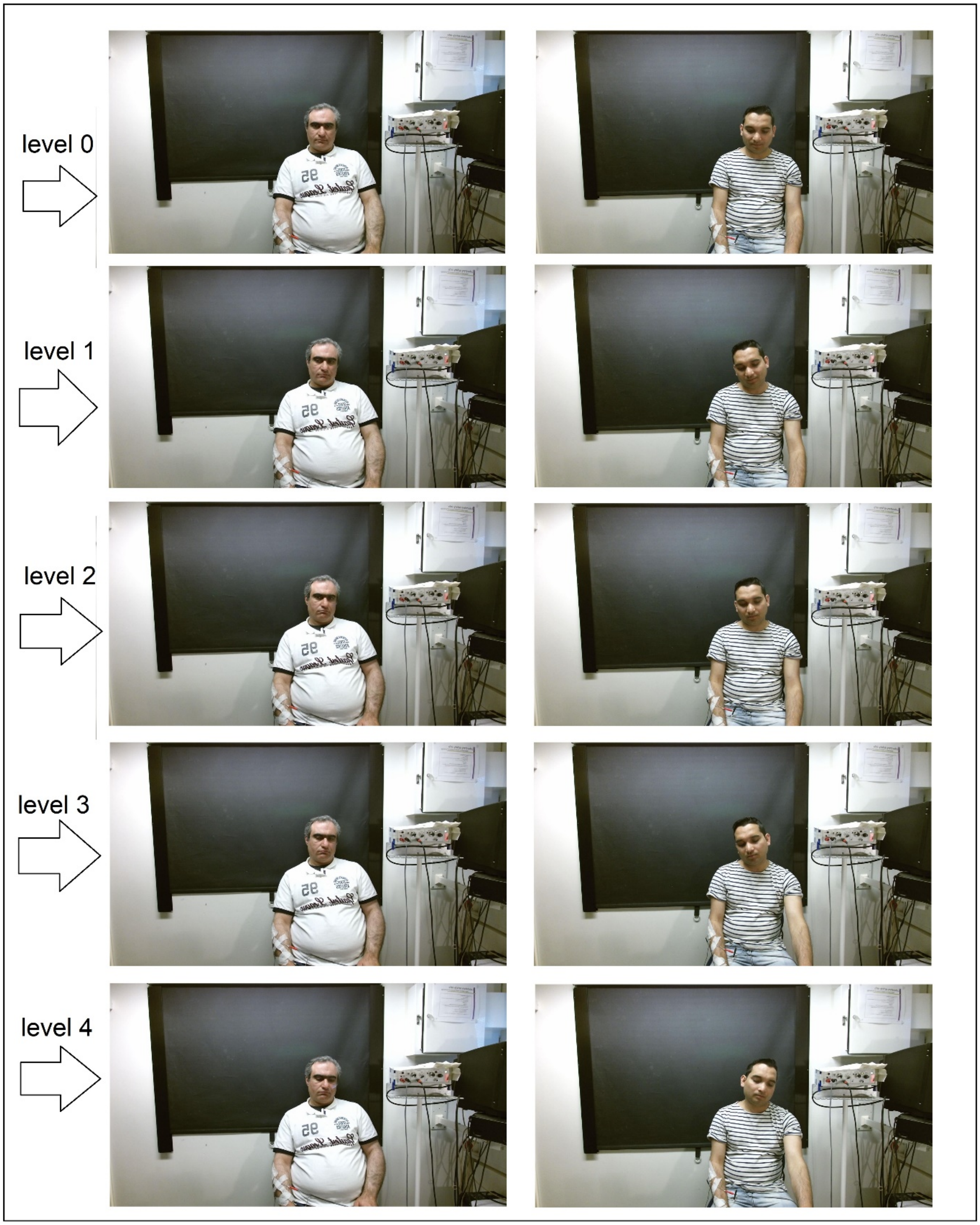

Fig. 6. Samples of selected dataset of MIntPAIN database [2, 16].

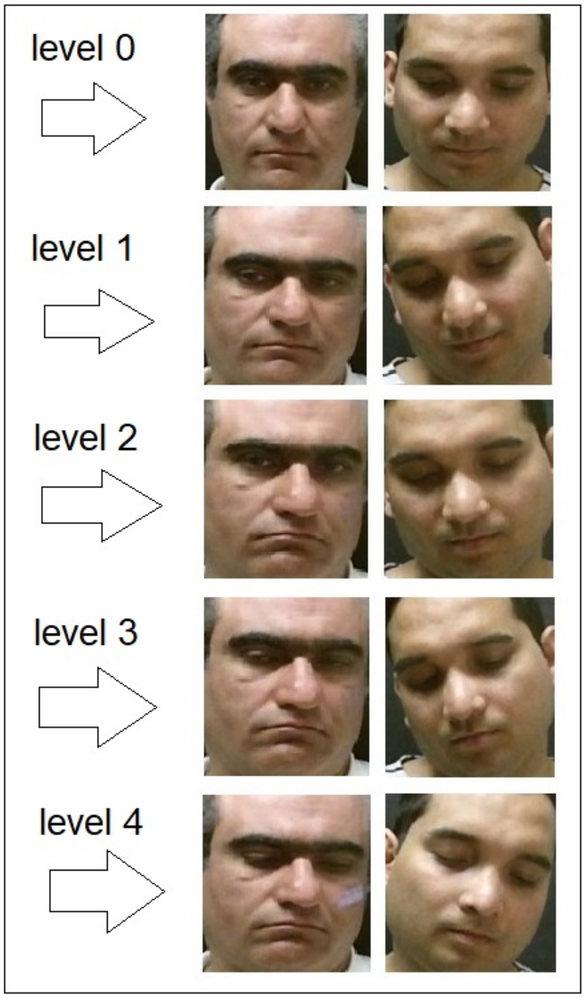

Fig. 7. Examples of video frames per 5 level after removing backgrounds, cropping, and resizing.

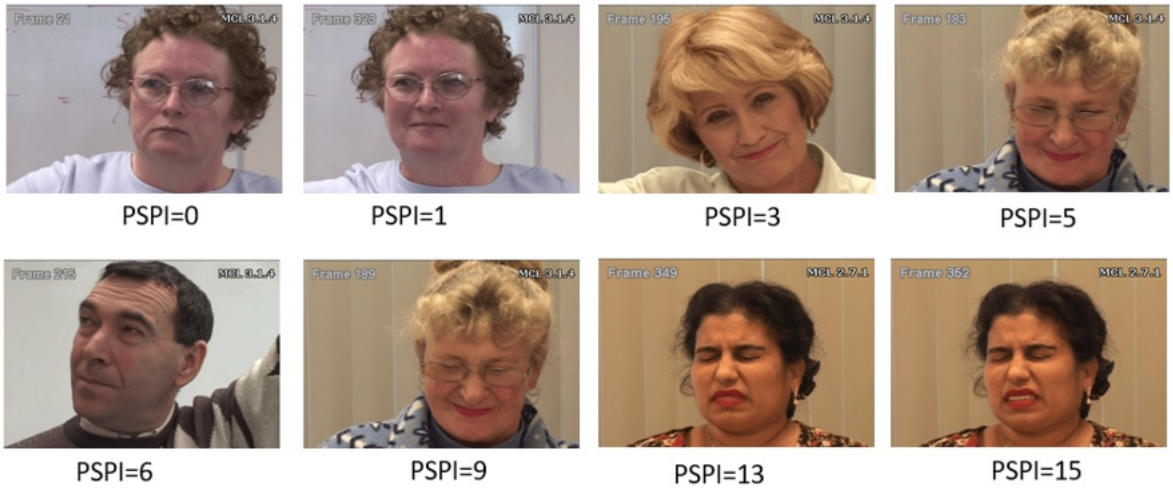

Fig. 8 Image frame samples of the UNBC-McMaster Shoulder Pain Achieve database [17].

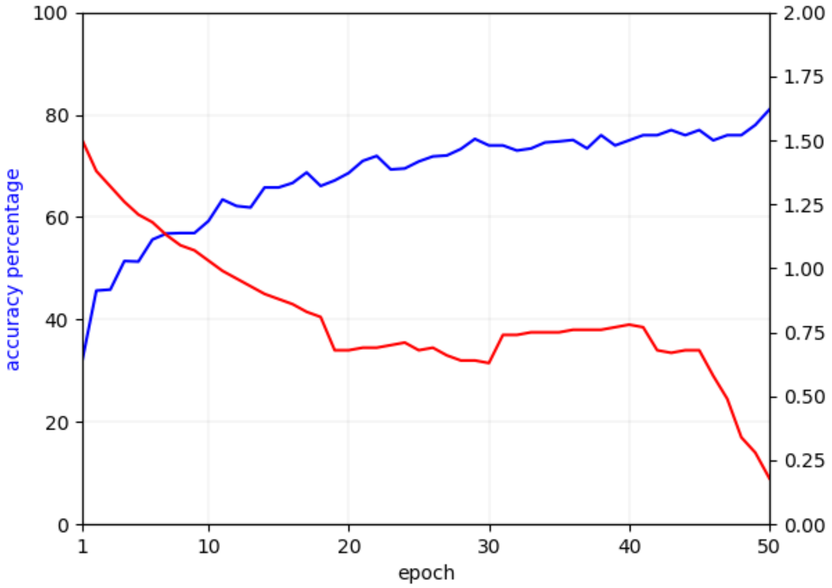

Fig. 9. Accuracy and loss error during 50 epochs in the early fusion of the EDLM model in the MIntPAIN database.

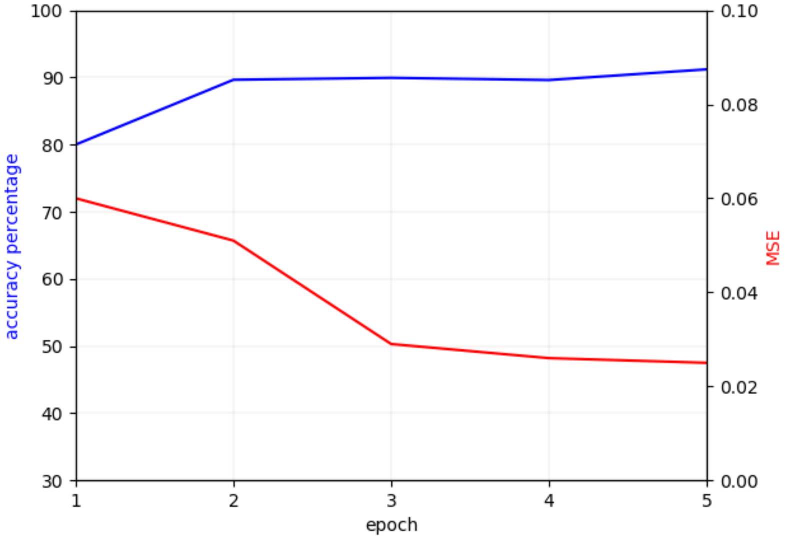

Fig. 10. Accuracy and MSE during 5 epochs in the late fusion of the EDLM model in the MIntPAIN database.

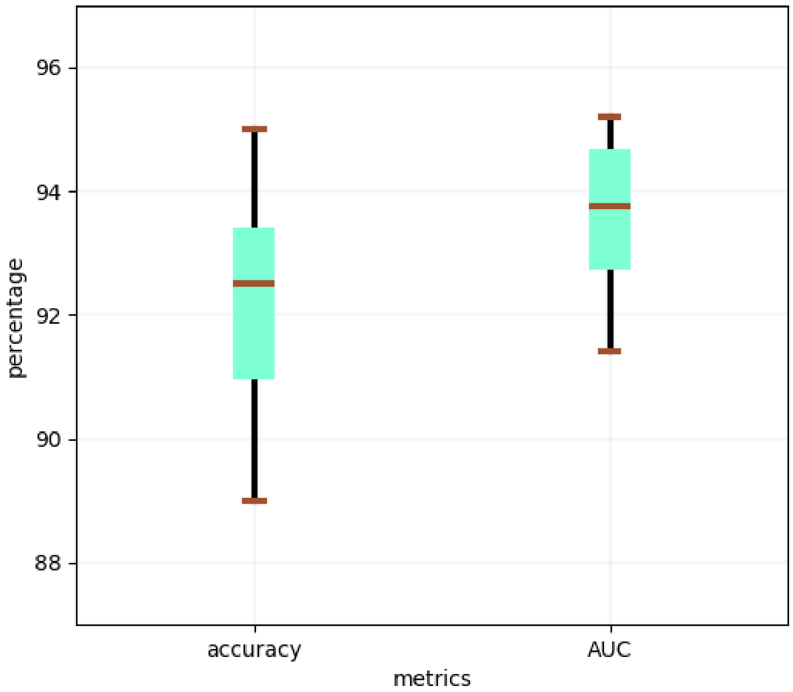

Fig. 11. Box plots of Accuracy and AUC for the proposed EDLM model in the MIntPAIN database.

Table 1. Properties of DNN1, DNN2, and DNN3 proposed in the late fusion stage.

DNN	Convolution layer 1	Convolution layer 2	RNN

DNN1	type = conv2d,
	filter number = 256,
	activation = ReLU,
	input shape = (1,5)
		type = conv2d
		filter number = 256,
		activation = ReLU,
		input shape = (1,5)
			type = BiLSTM,
			filter number = 256,
			dense = 4096,
			drop out = 0.5,
			activation = ReLU

DNN2	type = conv2d,
	filter number = 128,
	activation = ReLU,
	input shape = (1,5)
		type = conv2d
		filter number = 128,
		activation = ReLU,
		input shape = (1,5)
			type = BiLSTM,
			filter number = 32,
			dense = 4096,
			drop out = 0.5,
			activation = ReLU

DNN3	type = conv1d,
	filter number = 256,
	activation = ReLU,
	input shape = (1,5)
		None	type = BiLSTM,
			filter number = 128,
			dense = 4096,
			drop out = 0.5,
			activation = ReLU

Table 2. The average performance, best result, and worst results of the proposed model (EDLM) on MIntPAIN database for 10-fold cross validation.

Results	MSE	MAE	Accuracy	AUC
Average	0.0245	0.0341	92.26%	93.67%
Best	0.02102	0.028	95%	95.2%
Worst	0.03056	0.039	89%	91.4%

Table 3. Average pain level per five classes based on accuracy, f-score, precision, AUC metrics in the MIntPAIN database.

Metrics	No pain	Pain 1	Pain 2	Pain 3	Pain 4
AUC	87.3%	84%	85%	89%	91%
Precision	85.2%	85%	83%	88%	88%
f-score	86%	82%	82.2%	86.2%	90%
Accuracy	92.4%	89%	88%	93%	92%

Table 4. The average performance of the proposed model (EDLM) in the UNBC-McMaster Shoulder Pain database for 10-fold cross validation.

MSE	MAE	Accuracy	AUC
0.081	0.103	86%	90.5%

Table 5. The comparison of the obtained AUC and accuracy from the EDLM and the baseline model in the MIntPAIN database.

Classification models	AUC	Accuracy
VGGFace + 1 stream LSTM	87%	83.4%
The proposed EDLM model	*93.67%*	*92.26%*

Table 6. The time complexity of the proposed EDLM in compare with other baseline algorithm in the UNBC-McMaster Shoulder Pain database and MIntPAIN database.

Models	Database	Early fusion Time complexity (based on second) and number of applied epochs	Late fusion Time complexity (based on second) and number of applied epochs	Sum of the Time complexity
VGGFace + 1 stream LSTM	UNBC-McMaster	10400 / 5	560 / 5	10960
VGGFace + 1 stream LSTM	MIntPAIN	108000 / 50	1600 / 5	109600
VGGFace + PCA + 1 stream LSTM	UNBC-McMaster	5300/ 5	560 / 5	5860
VGGFace + PCA + 1 stream LSTM	MIntPAIN	40000 / 50	1600 / 5	41600
Proposed EDLM (VGGFace + PCA + 3 stream CNN-BiLSTM)	*UNBC-McMaster*	*5300 / 5*	*600 / 5*	*5900*
Proposed EDLM (VGGFace + PCA + 3 stream CNN-BiLSTM)	*MIntPAIN*	*40000 / 50*	*1700 / 5*	*41700*

Table 7. Comparing the proposed EDLM with the other state-of-the-art procedures in pain intensity recognition.

Ref	Pain Level	AUC (%)	Classifier	Accuracy (%)	MSE	Database	Data size
[17]	2	83.9	SVM	-	-	UNBC-McMaster	All
[27]	2	84.7	SVM	-	-	UNBC-McMaster	All
[31]	2	93.3	CNN-LSTM	83.1	0.74	UNBC-McMaster	Down-up
[16]	3	-	CNN-RNN	61.9	-	UNBC-McMaster	Down-up
[22]	2	-	-	-	1.54	UNBC-McMaster	16657 images
[2]	5	-	CNN-LSTM	32.40	-	MIntPAIN	All
[56]	4	98.4	PCA-CNN-RNN	91.2	0.04	UNBC-McMaster	Down-up
Proposed EDLM	*5*	*93.67*	*Ensemble CNN-RNN*	*92.26*	*0.0245*	*MIntPAIN*	*34800 images*
Proposed EDLM	*5*	*90.5*	*Ensemble CNN-RNN*	*86*	*0.081*	*UNBC-McMaster*	*10783 images*

Algorithm 1: The proposed EDLM algorithm

```
Procedure EDLM (input, n, j, batch)
        Pre-process (input)
        for k ← 0, n do
                finetune (VGG-Face)
                for epoch ← 0, j do
                        features ← train (finetune (VGG-Face))
                end for
                SF ← PCA (features)
                GN ← Calculate (GN)
                for epoch ← 0, j do
                        o1 ← DNN1(SF)
                        o2 ← DNN2(SF)
                        o3 ← DNN3(SF)
                        out ← merge (o1, o2, o3)
                        out ← GN (48)
                        train (model (SF, out))
                end for
        end for
end procedure
```

Declaration of interests

▯ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: