ENHANCED DEEP LEARNING PREDICTIVE MODELLING APPROACHES FOR PAIN INTENSITY RECOGNITION FROM FACIAL EXPRESSION VIDEO IMAGES

A Thesis submitted by

Ghazal Bargshady, M.Sc.

For the award of

Doctor of Philosophy

2020

Abstract

Automated detection of pain intensity from facial expressions remains a significant challenge in medical diagnostics and health informatics for providing a more intelligent pathway for the treatment of disease. Artificial intelligence methodologies, that have the ability to analyze facial expression images, utilizing an automated machine learning algorithm, can be a promising approach for pain intensity analysis. As a rapidly emerging machine learning technique, deep neural network algorithms have made significant progress in both feature identification, mapping, and modelling of the pain intensity from human facial images, with a strong potential to aid the health practitioners in the diagnosis of certain medical conditions from observable symptoms and signs of disease. While there is a significant amount of research within the pain recognition and management area that adopts facial expression datasets into deep learning algorithms to detect the pain intensity in binary classes, and identifying the pain and non-pain faces, the volume of research in identifying pain intensity levels in multi-classes remains rather limited. Although the effectiveness of deep learning models has been demonstrated, obtaining accurate algorithms to automatically detect pain in multi-class levels is still a challenging task and needs major improvement in the predictive skill of such techniques. In addition to this challenge, there exists individual behaviors, such as smiling or crying in pain situations by some patients that can make it potentially more difficult to measure the actual pain arising from a disease condition using the patient's facial expressions through deep learning models.

The PhD Thesis reports on the design, statistical validation and the practical testing of new enhanced deep neural-network algorithms tailored for the effective and efficient detection of pain intensity in humans by means of using a facial expression video image. To explore the robustness of the proposed deep learning algorithms, reliable information sourced from the UNBC-McMaster Shoulder Pain Archive Database, and the MIntPAIN database, comprised of human facial images, were used for training and testing of the proposed pain classification model. To provide enhanced model performance, the models were coupled with the fine-tuned VGGFace pre-trainer as a feature extraction ancillary tool. To reduce the dimensionality of the classification model input dataset and to extract the most relevant facial features in modeling the pain intensity, the Principal Component Analysis (PCA) was applied to improve its

computational efficiency. The pre-screened facial image features, used as potential model inputs, were then transferred to generate the newly enhanced deep learning models. In this project, three variants of the enhanced deep learning-based classifier algorithms were developed and evaluated , including the joint hybrid CNN-BiLSTM (EJH-CNN-BiLSTM) algorithm, the ensemble deep learning model (EDML), and a temporal neural network (TCN) with the Hue, Saturation, Value (HSV) color space as (HSV-TCN) algorithm. All algorithms were tested on human facial image dataset to model pain intensity.

The EJH-CNN-BiLSTM deep learning algorithm comprised of convolutional neural networks, linked to the joint bidirectional-long-short-term memory (BiLSTM), for multi-classification of human pain. The resulting EJH-CNN-BiLSTM classification model, tested to estimate four levels of pain, revealed high accuracy (90%) and AUC (98.4%) on the balanced UNBC-McMaster Shoulder Pain database, benchmarked by a diverse suite of model performance evaluation indicators. The proposed classifier was improved by applying in a stacked ensemble deep learning model (EDLM). This ensemble deep learning model has three deep learning models based on CNN-LSTM and their output were merged to classify 5 levels. The results show the model accurately classifies pain to identify multi classes of pain level and its performance is high in compare with other baseline models and the state-of-the-art methodologies. The accuracy reached to 86 % and AUC of 90.5% for UNBC-McMaster Shoulder Pain database and AUC of 93.67% and accuracy of 92.26% for MIntPAIN database.

Although the proposed models outperform pain detection from facial images in multi levels, the speed of the algorithm need improvement. To speed up the deep learning based pain recognition systems from human facial videos' images a new algorithm based on the temporal convolutional network with HSV color space inputs was developed and the evaluation results shows its effectiveness and efficiency of it is noticeable in compare with other models. The obtained results show accuracy of 94.14% and AUC of 91.3% in UNBC-McMaster Shoulder Pain database and accuracy 89% and AUC 92% in MIntPAIN database for 5 classes and the algorithm run 6 times faster than the above models.

In summary, the results from these experiments clearly prove that the proposed deep learning approaches were able to generate accurate performance for the recognition of

pain intensity levels from the videos' images of facial expressions and could be adopted in health care systems. The newly developed techniques provide key contributions to health informatics area, as prominent artificial intelligence tools to evaluate a patient's pain level more accurately that manual methods. Subsequently, these techniques could be applied in the management and treatment of pain in patients by using a more coherent, accurate, and effortlessness methodology.

Certification of Thesis

This Thesis is entirely the work of Ghazal Bargshady except where otherwise acknowledged. The work is original and has not previously been submitted for any other award, except where acknowledged.

Principal Supervisor: Prof. Jeffrey Soar

Associate Supervisor: Dr. Xujuan Zhou

Associate Supervisor: Associate Professor Ravinesh C Deo

External Supervisor: Dr. Frank Whittaker

External Supervisor: Prof. Hua Wang

Student and supervisors' signatures of endorsement are held at the University.

Acknowledgments

I would like to express my appreciation to my supervision team including Professor Jeffrey Soar, for offering me the opportunity to pursue a PhD degree in the Artificial Intelligence and Image Processing research field at University of Southern Queensland (USQ). During the process of pursuing my PhD, Prof. Soar has given me kind advice and support, which has been crucially important for my PhD study. I would like to express gratitude to my associate supervisors, Dr. Xujuan Zhou and Assoc Prof. Ravinesh C Deo, for your expert advice and encouragement throughout the hard times of my PhD. Without your careful supervision, the finalization of this thesis would not be possible. I would like to express my gratitude again to Dr. Xujuan Zhou who helped me with my technical knowledge, designing the methods, and presenting the results. I am so fortunate to have Assoc Prof Ravinesh C Deo as my supervisor who actively helped me with designing the methods and writing articles and provide in-depth feedbacks to improve my work. I would like to express gratefulness to my external supervisors including Dr. Frank Whittaker and Prof. Hua Wang. I greatly appreciate Dr. Whittaker as the funder and industry partner of the ARC Linkage grant and Prof. Hua as an academic sponsorship of this grant who supported financially me during PhD study.

This research has been supported by an Australian Government Research Training Program Scholarship through the Australian Research Council (ARC) grant LP150100673, Nexus eCare and the University of Southern Queensland. I am thankful to the School of Management and Enterprise and all my fellow graduate students for their support throughout my PhD journey. I am grateful to Dr. Douglas Eacersall for his writing advice of this thesis.

Last but no means least, I express my sincere gratitude to my family and friends for their support and patience. My heartful thanks to my father and my sister, for their encouragement, patience, and love from miles away.

Keywords

Deep learning, facial expression, pain recognition, image processing, computer vision, data science, health informatic, pain intensity estimation, automated pain detection, feature extraction, transfer learning, temporal convolution networks, convolutional neural networks, recurrent neural networks, artificial intelligence, expert systems.

Australian and New Zealand Standard Research Classification (ANZSRC)

080199 Artificial Intelligence and Image Processing not elsewhere classified (70%)
080309 Software Engineering (30%)

Fields of Research (FoR) classification

GROUP 0801 ARTIFICIAL INTELLIGENCE AND IMAGE PROCESSING (70%)
GROUP 0803 COMPUTER SOFTWARE (30%)

Table of Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

No	Abbreviation	Description
1	AAM	Active Appearance Models
2	AFER	Automated Facial Expression Recognition
3	AI	Artificial Intelligence
4	AIDS	Acquired Immunodeficiency Syndrome
5	AR	Action Research
6	ARC	Australian Research Council
7	ASM	Active Shape Model
8	AUC	Area under Curve
9	BCP	Base Classifier Pool
10	BiLSTM	Bidirectional Long Short Memory
11	CDF	Cumulative Distribution Function
12	CNN	Convolutional Neural Network
13	CRF	Conditional Random Field
14	CV	Cross Validation
15	DSL	Deep Structured Learning
16	EDLM	Ensemble Deep Learning Model
17	EJH	Enhanced Joint Hybrid
18	ELS	Ensemble Learning System
19	FACS	Facial Action Coding Systems
20	FER	Facial Expression Recognition
21	FN	False Negative
22	FP	False Positive
23	FPR	False Positive Rate
24	FPSR	Faces Pain Scale-Revised
25	GCN	Global Contrast Normalization
26	GDF	Geometric Distance Feature
27	GWF	Gabor Wavelet Filter
28	HCRF	Hidden Conditional Random Field
29	HE	Histogram Equalization

30	HOG	Histogram of Oriented Gradient
31	HSV	Hue, Saturation, Value
32	ICA	Independent Component Analysis
33	KNN	*k*-Nearest Neighbor
34	LBP	Local Binary Pattern
35	LBP-TOP	Local Binary Pattern – Three Orthogonal Planes
36	LOOCV	Leave One Out Cross Validation
37	LSTM	Long Short-Term Memory
38	MAE	Mean Absolute Error
39	MIL	Multiple Instant Learning
40	MIntPAIN	Multimodal Intensity Pain
41	MNF	Minimum Noise Fraction
42	MSE	Mean Squared Error
43	NRS	Numerical Rating Scale
44	OvO	One-vs-One
45	OvR	One-vs-Rest
46	PCA	Principal Components Analysis
47	PR curves	Precision-recall curves
48	RNN	Recurrent Neural Network
49	RVM	Relevance Vector Machine
50	PSPI	Prkachin and Solomon Pain Intensity
51	RGB	Red Green Blue
52	ROC	Receiver Operating Characteristics
53	SD	Standard Deviation
54	SVM	Support Vector Machine
55	SGD	Stochastic Gradient Descent
56	TCN	Temporal Convolutional Network
57	TN	True Negative
58	TP	True Positive
59	TPR	True Positive Rate
60	VAS	Visual Analogue Scale
61	VRS	Verbal Rating Scale

CHAPTER 1

Introduction

This doctoral research thesis aims to develop enhanced deep learning predictive modelling techniques for pain intensity recognition from facial expression images. Automatic pain detection technology is essential for healthcare provider special for patients with limited communication ability or required regular symptoms' reports. Facial expression is one of the most meaningful and natural ways to interpret pain and emotional states. Machine learning algorithms, implemented as an artificial intelligence predictive system, can offer an alternative mechanism for pain assessment task to detect both the pain and its relative intensity level from facial expression images. While the effectiveness of deep learning models and computer vision technology is demonstrated, in the facial data domain obtaining accurate algorithm to detect pain in multi levels still is challenging and needs further improvement. Enhanced deep neural networks were developed and evaluated to detect pain suffered by patients based on their facial expression images.

In this chapter, the key background information regarding pain recognition systems from facial expression images are provided. Following this, automated pain recognition systems developed through artificial intelligence methodologies are explained. Deep learning, as a recent artificial intelligence technique applied in pain intensity detection from facial expression images, is introduced and its problems and challenges in automated pain detection systems from facial expression is explained. This is followed by the research questions. The research aims and objectives, including the research significance are elaborated on in the following sections, and finally, the chapter is closed with an outline of the thesis organization.

1.1 Background of Pain Recognition Systems

Pain is an individual experience and a mental sense that is considered as a relatively complex phenomenon and is not yet well understood. Pain is an unpleasant sensory and emotional experience associated with actual or potential tissue damage (Aydede 2017). Pain is classified into two categories, mainly as an acute or a chronic pain

(Aydede, 2017). Acute pain is associated with a health condition, a medical diagnosis, or a surgical procedure (Aydede, 2017). However, chronic pain is associated with a high mortality disease such as cancer, Acquired Immunodeficiency Syndrome (AIDS), Parkinson's Disease, low back pain, failed back surgery, or chronic headache (Payne, 2000). Chronic pain can be caused by the disease itself as a symptom or as a treatment progression (Payne, 2000). The treatment of chronic pain should be self-monitored to access information by the pain management teams, such as type of pain and the level of complexity (Lynch, 2011). An effective and consistent pain assessment is required to select the suitable treatment, and to modify the treatment, and monitor the patient's pain status.

In clinics, several techniques are applied to measure pain, including but not limited to the Hierarchy of Pain Assessment Techniques, Search for Potential Causes of Pain, Observe Patient Behaviors, and Proxy Reporting (Herr et al., 2011). A hierarchy of assessment techniques is generally recommended for use in medical clinics, and the following steps can be used as a template for the initial assessment and treatment procedure (Herr et al., 2011; Werner et al., 2019):

a. Obtain a patient self-reporting record and check its feasibility, otherwise, a described document is required to elaborate on the incapability reasons of the self-report system. The commonly used self-reporting pain-rating scales consist of Visual Analogue Scale (VAS), Numerical Rating Scale (NRS), and Verbal Rating Scale (VRS) (Martinez et al., 2017).
b. Identify the cause of pain such as pathologic conditions. If there is a lack of self-reports and behavioral signs, pathologic conditions such as surgery, trauma, osteoarthritis, wound, history of persistent pain, blood draw, heel sticks can be applied.
c. List patient behaviors or use a behavioral assessment tool that may indicate pain. When a self-report is absent or the amount of information therein is limited, explanations for the deficiencies regarding the self-report and advance investigations and observations are required. In this case, the observation of human behavior which considers facial expressions, or vocalizations, or body language is taken as a valid method for pain management. Some scales are designed and validated for patients based on behavior observation including NIPS, CRIES, FLACC for infants and PACSLAC, DOLOPLUS2, PAINAD

for elderly people with severe dementia, and BPS, CPOT, NVPS for critical patients.

d. Identify information from caregivers, family members, parents in a reliable proxy reporting about the patient's behaviors. Caregivers and family members should be actively encouraged to contribute to the pain assessment procedure.
e. Try a pain assessment self-report from a patient with inadequate communication and cognitive skills such as finger span or eye blink to answer yes or no questions.

However, there is some degree of limitation with these rather manual pain assessment measures, especially regarding the self-reporting elements of pain management systems. These tools cannot always be used with infants, patients with certain types of neurological impairments and dementia, patients requiring postoperative care. In clinics using self-reported pain, measurement is not possible in many cases and even though clinical pain measurement can be undertaken frequently by medical staff and nurses such regular pain measurement is not cost-effective (Werner et al., 2019). This method may also mean that some relevant pain sensations may be missed or misread by a clinician (Werner et al., 2019).

Since facial expressions, as a behavioral observation method of pain recognition, have a significant role in communicating and managing the pain, scientists have reverted their interests to developing machine learning algorithms to train systems to decode complicated associations between facial expressions and pain (Liu et al., 2018). Compared with humans, machine learning algorithms can utilize many different facial features including landmarks, colors, lighting, and movements to detect human pain and emotion (Liu et al., 2018). In a practical sense, faces can carry important information for any communal interactions, including the expression of emotions and pain. Significant efforts are extended to identify reliable and valid facial indicators of pain. In recent years, considerable progress made in the machine learning area to automatically recognize the facial expressions related to pain and emotions. Researchers have therefore applied machine learning algorithms to undertake the challenging task of computerized pain detection in healthcare for patients suffering pain issues (Bartlett et al., 2014).

Facial expression recognition is a very challenging task since the faces were presented in different poses, varying with respect to the different ages of any patient. To deal

with these challenges faced by the healthcare industry, extracting the features from such complex characteristics of patients' faces plays a key role in facial expression recognition (FER) (Xu et al., 2015), and the corresponding pain intensity. As is demonstrated in Fig. 1-1 the general framework for facial expressions consists of the following steps: data acquisition, data pre-processing, feature extraction, image classification, and post-processing, including any modelling tasks to make important decisions about the use of the facial expression recognition tool (Chibelushi & Bourel, 2003).

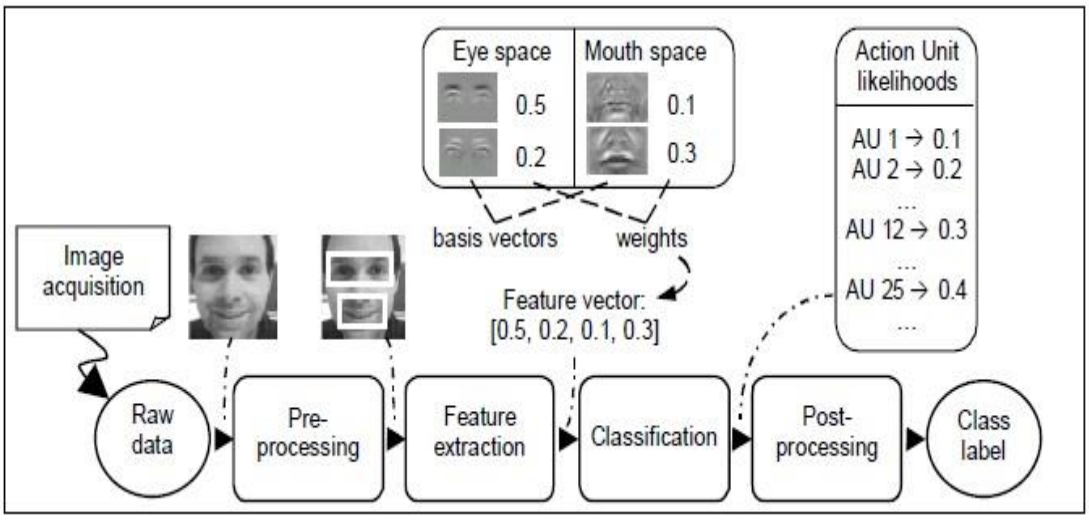

Fig. 1-1. A schematic view of the general Facial Expression Recognition (FER) framework used in a practical healthcare environment (Chibelushi & Bourel, 2003).

The FER systems were categorized into two main groups, is based on the feature representations: static image FER and dynamic sequence FER. In the first method, the feature representation is encoded with only spatial information from the single image, whereas for the dynamic method the temporal relation among sequential frames in the input facial expression series, is considered. To attain an accurate and a versatile modelling platform for any facial image recognition task, it is important to have enough labelled training datasets to provide multiple and varied populations and environments for designing a deep facial recognition system. Ekman and Friesen (1978) developed the Facial Action Coding System (FACS) which provides an objective meaning for measuring the facial muscle contractions in facial expression (see Table 1-1 and Fig. 1-2). FACS developed for researchers to measure the activity of facial muscles from facial images. Each element of facial movement is called an

Action Unit (AU) (Chibelushi & Bourel, 2003). FASC includes 46 AUs and it produces facial features which can be identified in the image (Ekman & Friesen, 1978).

Table 1-1. Action units measured in (Ekman & Friesen, 1978)

AUs	**Muscle Involved**	**Description of Action**
1	Inner frontalis	Raises inner corner of brow
2	Outer frontalis	Raises outer corner of brow
1+2	Frontalis	Raises entire brow
4	Corrugator Procerus	Lowers and pulls brows together
6	Orbicularis oculi, outer portion	Squints eyes, makes crowfeet wrinkles
7	Orbicularis oculi, inner portion	Squints eyes, raise and straightens lower lid
9	Levator labii superioris, alaque nasi	Wrinkles nose
10	Levator labii superioris	Raises upper lip
12	Zygomatic major	Common smile
14	Buccinator	Dimples cheeks
15	Triangularis	Lower corners of lips
16	Depressor labii inferiors	Pulls lower lips down
20	Risorius	Stretches lip corners straight to the side
45	Orbicularis oculi	Blink or wink

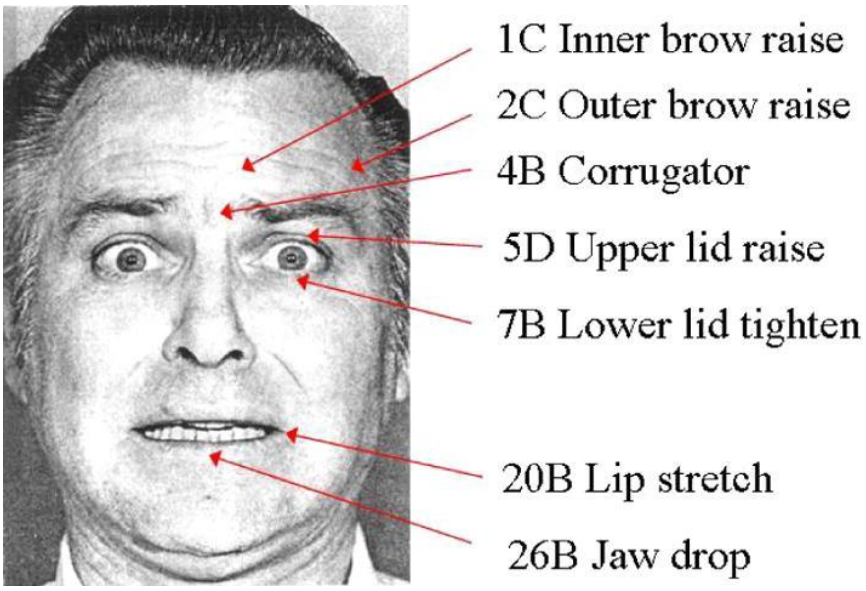

Fig. 1-2. Example of facial action decomposition from facial action coding system (Littlewort et al., 2009).

Prkachin and Solomon (2008) measured the pain based on AUs and found that actions such as AU4, AU6, AU9, AU10, and AU43 contain most of the information about facial pain. It is noteworthy that the Prkachin and Solomon Pain Intensity (PSPI) pain scale is currently the only quantitative metric that can define the pain on a frame-by-frame basis. PSPI defined pain as the sum of the intensities of brow lowering, orbital tightening, levator contraction and the eye closure. Accordingly, PSPI metric defined as follows (Prkachin & Solomon, 2008):

$$Pain = AU4 + (AU6 \text{ or } AU7) + (AU9 \text{ or } AU10) + AU43 \quad (1\text{-}1)$$

After labelling the raw image dataset, the process of capturing facial images is essential since the potential sources of errors can encounter during the data collection process. This can affect the credibility of the actual database inducted by exogenous errors that, if not addressed appropriately, can potentially lead to false data features in an automated facial image and pain recognition system. The potential problems of image processing include segmentation, deformation, illumination, affordance, and viewpoints complications. Illumination or poor lighting in digital images, or a deformation of images due to a low-quality photo occur in the image analysis process. If these issues are not resolved carefully, the accuracy of the image recognition algorithm is expected to deteriorate, leading to significant errors in the result, making the algorithm virtually useless. The preparation of the image data as a prior task in

image processing technique helps resolve these challenges by reducing noise, filtering, and enhancing the contrast to clarify the features. A pre-processing of all input data before designing a robust model is an important task advocated by previous works (Haque et al., 2018; Rodriguez et al., 2017; Zhou et al., 2016). For example, Rodriguez et al. (2017) in a pain detection task from facial video data, at first, cropped and frontolyzed the faces' video frames.

Data normalization is a critical preprocessing technique that can change the range of a pixel's value in images with a poor contrast. Normalization, which is sometimes called contrast stretching or histogram stretching, in data processing such as digital signal processing or image processing refers to the dynamic range expansion. Face analysis, as a part of image processing, is complex due to the changes in the appearance of the face caused by posing variations and illumination changes. Illumination normalization algorithms, discrete cosine transform based, difference of Gaussian such as Global Contrast Normalization (GCN), and local normalization and histogram equalization are used frequently as normalization techniques in facial expression recognition (Kuo et al., 2018; Pitaloka et al., 2017). Selecting and applying the right normalization can be a significant factor in getting the model to train effectively. Using normalization in the deep learning algorithms significantly improves its performance on image classification. Warping techniques as a normalization method based on the center positions of the distinctive facial features such as the eyes, nose and mouth, are helpful in managing rotated faces where their facial expressions are misrepresented or may become partially invisible in contrast to frontal face displays. As a solution to this problem, Zhou et al. (2016) warped every facial image in Red Green Blue (RGB) channels individually, then merged all channels to obtain the final RGB warped faces.

Machine learning-based pain assessment, if developed appropriately, is expected to be more accurate and less biased compared with human observations and its scalability is priceless for clinical utilizations (Liu et al., 2018). A wide range of machine learning methods were developed for automatic pain prediction from human facial expression. These machine learning methods used in previous studies can include many different general categories of models, such as, but not limited to linear regressions, naive Bayes, logistic regression, support vector machine, gaussian processes, random forests, genetic algorithms and deep neural networks (Liu et al., 2018). Deep neural networks have become an attractive classifier algorithm of choice in many machine

learning tasks. However, most of the recent work in the facial expression area involves the use of deep learning and neural networks due to their superior performance (Liu et al., 2018).

1.2 Research Problems

Machine learning algorithms, implemented as an artificial intelligence predictive system, can offer a viable alternative mechanism for pain assessment tasks, such as using a camera to collect the relevant data (*e.g*., a face image), to detect both the pain and its relative intensity. Facial expression is one of the most meaningful and natural ways to interpret pain and emotional states. The challenge in the facial expression recognition lies in the large visual feature variations caused by person-specific characteristics of expressions and variations from extrinsic conditions such as illumination and the view point (Zhang et al., 2017). Another key challenge of facial-expression recognition is to develop effective representations to balance the complex distribution of intra- and inter- class variations (Liu et al., 2017). There are noticeable differences in individual patient's faces such as varying their poses and their age differences. There are some different characteristic behaviors in individuals that can affect facial expressions such as smiling or crying in pain which make measuring pain, from facial expressions using deep learning models, more difficult. The specificity, sensitivity, and the effectiveness of the deep learning algorithms still require improvements through enhancing the existing deep learning techniques and modifying the accuracy of current models.

As discussed in section 1.1, the commonly used self-reporting pain-rating scales in clinics were VAS, NRS, and VRS (Martinez et al., 2017) which measure pain in nominal and multi-class method. Although, pain is ordinal data and could be classified by ordinal regression or ordinal classification, in this research work an automatic pain detection approach in multi-class is designed by applying deep learning to simulate the most common self-reported pain detection systems such as VAS and NRS techniques from patients' facial expressions which are multi-class and nominal measurement methods. This technique would be useful for healthcare providers and patients who has difficulty in communicate pain by self-reporting systems due to physical condition such as certain types of neurological impairments and dementia, patients requiring postoperative care.

In pain recognition tasks, the major issue and challenge is availability of databases. Collecting data and sharing information from clinical real-use cases is difficult. The number of available public databases including painful facial images and video frames is very limited. An ideal dataset should be multimodal with more descriptions, high quality information about labelling, with relevant information to improve comparability of results. Performing the deep learning modelling experiment in most public available pain databases is challenging due to imbalanced data labels, and poor quality of image data such as unfair environment brightness, shooting angle and distance, background, and noise.

The following research questions were developed and addressed in this PhD study:

RQ 1: *What are the most recent deep learning model advancements in pain recognition from facial expressions?*

RQ2: *What is the most effective deep learning algorithm to extract and select features from facial pain images?*

RQ3: *What is the most effective and efficient facial expression pain recognition deep learning algorithm to classify pain intensity on multi-level?*

RQ4: *How effective are the developed enhanced deep learning models to recognize pain from facial expression*?

RQ5: *How efficient are the developed enhanced deep learning models to recognize pain from facial expression?*

1.3 Aim of the PhD Thesis

The research reported on in this PhD thesis aimed to develop, validate, and evaluate independently a set of new pain recognition deep learning models that were adopted to detect pain and its intensity from human facial expressions as video recorded images. These models were developed and evaluated for improving existing deep learning pain analysis techniques from facial behavior and tackling the above-mentioned problems. The PhD thesis provides multiple solutions to resolve the challenges within deep learning pain recognition systems using facial expressions. After investigating the other researchers' works, new developments are proposed and

verified extensively by means of advanced statistical score metrics and a visual analysis of results.

This research work, undertaken to resolve the problems mentioned in the previous section, results in the following original contributions:

1. Development of a new deep learning feature extraction model by finetuning a pre-trained convolutional neural network and reducing features by principal component analysis.
2. Development of a joint hybrid classification model base on convolutional neural networks and recurrent neural networks.
3. Development of a staking ensemble deep learning model based on convolutional neural network and recurrent neural networks by extending the classifier introduced in original contribution number 2.
4. Development of a deep learning classifier by modifying temporal convolutional network architecture and adjusting the input images' colour spaces to improve and enhance deep learning pain recognition from facial images task.

Three pain detection algorithms were developed and evaluated to detect pain from facial expression video frames by applying the new feature extraction model. The results, reported in other sections of this thesis, show that the enhanced pain recognition algorithms have performed effectively, and efficiency as demonstrated by evaluating accuracy and measuring speed.

1.4 Significant Contributions

It is essential to assess, reassess, and document pain routinely and quickly to simplify treatment procedures and the interaction among health-care services (Herr et al., 2011). Deep learning algorithms play a role in helping healthcare providers to achieve more accurate and efficient assessment of pain intensity level could monitor pain continuously. Medical staff require an accurate, regular, timesaving, and secure AI tool to increase their estimation in illnesses diagnosis, facilitate patients' treatment procedures, and keep healthcare workers safe from infectious diseases.

Deep learning approaches have become a mainstream machine-learning technique with capacity in various nonlinear modelling tasks such as the classification and

feature extraction process from complex datasets especially in healthcare domains. Transfer learning and pre-training techniques of deep learning were identified as successful techniques in extracting features. While the effectiveness of deep learning models has been demonstrated, in the facial data domain obtaining an accurate algorithm to detect pain in multi levels is still challenging and needs improvement (Parkhi et al., 2015; Walecki et al., 2017).

This research work undertaken to resolve the problem discussed in Section 1.2, and answer the research questions, results in the following outcomes:

1. A new feature extraction, based on transfer learning, that is designed to extract the most important features by using a fine-tuned VGGFace and the PCA dimension reduction method.
2. A new and effective enhanced joint hybrid deep learning classifier based on CNN-RNN that automatically estimates pain levels from facial expressions.
3. The enhanced joint hybrid deep learning improves by extending into a new developed, stacked ensemble deep learning CNN-RNN model which is a more effective and accurate pain level classifier using facial expression video images.
4. A new deep learning algorithm constructs based on temporal neural networks with HSV color space inputs to speed up the deep learning pain detection algorithm from video frames and evaluated as an effective and efficient pain recognition algorithm when compared with other benchmark models.
5. Pain detection frameworks that can be easily implemented as an artificial intelligence algorithm in healthcare platforms such as mobile applications or web portals, to manage patients' pain levels automatically and thus be practically applicable for pain detection tasks.

1.5 Thesis Organization

This thesis is organized as followings:

Chapter 1 presents the background and problem statement of the current research. The chapter starts with an introduction about pain detection histories in the clinic, and the necessity of automatic pain detection tools. Additionally, the challenges and problems in the deep learning pain detection algorithm from facial expressions were discussed

and the thesis research questions identified. The aims and original contributions of the thesis are also outlined in detail.

Chapter 2 overviews the different deep learning pain detection models from human facial images. A summary of the literature review covering automated pain recognition from facial expressions, image pre-processing, deep learning feature extraction techniques, deep learning classifiers, and related databases is presented. The findings of this chapter provide information to answer to research question 1 and lead to identify problems, develop new enhanced pain recognition deep learning models from facial data, and find suitable database to train and evaluate the proposed algorithms.

Chapter 3 discusses the research methodology developed in this thesis. A research methodology is a way in which one proceeds to solve the problem and a description of how the research will be conducted. The proposed research methodology in this thesis combines the strength of both the scientific approach and action research to achieve the research objectives. A framework is introduced for automatic pain detection from facial video images. This framework shows the phases and relation between phases. Then the applied datasets for training and testing the proposed algorithms, applied evaluation metrics and validation approaches, and experimental configuration are explained in this chapter.

Chapter 4 presents the Enhanced Joint Hybrid (EJH-CNN-BiLSTM) proposed models to detect pain and its intensity from facial expression video images. The EJH-CNN-BiLSTM is a new model proposed in this thesis which is recognize pain level from facial expression effectively and accurately. In this chapter the proposed model and its components including image preprocessing techniques applied in this model, the newly designed feature extraction algorithm, and the newly developed hybrid joint deep learning classifier are elaborated and the obtained results and comparison with the state-of-the-art models' results are discussed.

Chapter 5 explains the newly developed Ensemble Deep Learning Model (EDLM) proposed models to detect pain and its intensity from facial expression video images. In this chapter the proposed model in Chapter 4 as EJH-CNN-BiLSTM is extended in an ensemble deep learning approach to examine its effectiveness for pain detection task. The obtained results show its accurateness level is high in comparison with the

state-of-the-are models, and other base line models. The obtained results, evaluation, and comparison are discussed.

Chapter 6 presents a significantly improved version of the Temporal Convolutional Networks (TCN) algorithm with Hue, Saturation, Value (HSV) color space input data as (HSV-TCN) to detect pain and its intensity from facial expression video images. Although the proposed models in Chapter 4 and 5 outperform pain detection from facial images in multi levels, the speed of the algorithm need improvement. To speed up the deep learning based pain recognition systems from human facial videos' images a new algorithm based on the TCN deep neural network which is modified for this task with HSV color space inputs is developed and the evaluation results shows its effectiveness and efficiency of it is noticeable in compare with other two proposed and developed models and models presented in the literature review.

Chapter 7 concludes the findings of this thesis by presenting conclusions, limitations, and opportunities for future research.

CHAPTER 2

Literature Review

In this chapter, the necessary background and state of current methods will be detailed and reviewed. The review is organized in three main sections. Section 2.1 introduces feature extraction methods, and their challenges. A literature review has been undertaken to identify current feature extraction methods especially in pain detection tasks by applying deep learning methods. Section 2.2 gives background information of deep learning classifiers and methods were applied to pain detection from facial expressions. The literature reviewed in this section to identify the current deep learning algorithm for this task and recognize the challenges in multi classes pain detection by applying deep learning. Section 2.3 explains the available databases in pain detection area from facial expression images and videos and review their properties and limitations.

2.1 Deep Learning Techniques in Feature Extraction

The feature extraction methods are commonly applied before the classification task. There are some traditional methods for feature extraction such as Active Appearance Model (AAM), Active Shape Model (ASM), Geometric Distance Feature (GDF), Local Binary Pattern – Three Orthogonal Planes (LBP-TOP), Gabor Wavelet Filter (GWF), and Histogram of Oriented Gradient (HOG). Table 2-1 demonstrates a detailed set of information about each of the non-deep learning techniques.

Table 2-1. Traditional methods used for feature extraction.

Techniques	Description
AAM	AAM which contains a statistical model of shape and grey-level appearance which can be generalised to almost any face (Cootes et al., 1998; Edwards et al., 1998).
ASM	ASM algorithm is a statistical model of the image which deforms iteratively an object to fit it in a new image. It is a fast approach to matching a controlled set of points to a new image (Cootes et al., 1995).
GDF	GDF represents facial landmarks such as shapes and location including mouth, eyes, eyebrows, and nose. GDF is distances between certain points of the facial image (Mozaffari et al., 2010; Zhang et al., 1998).
LBP	LBP compare the centre pixel value with the neighbourhood pixel values by using a binary code which is generated by allocating the value one to the higher neighbour pixel value and zero to the rest. Binary code is converted to decimals to get the LBP value of the centre pixel (Ahonen et al., 2006).
GWF	GWF is as an appearance-based method is used as an image filter in all or part of the face to extract feature vectors and edge detection (Lyons et al., 1998).
HOG	HOG is a feature descriptor in computer vision and image processing and used for object detection. The input image divided into small spatial regions which are called cells (Dalal & Triggs, 2005).

The advent of Convolutional Neural Networks (CNNs) that are considered as powerful machine learning models, can achieve remarkable results in image classification and facial expressions problems, with the pre-trained CNNs being particularly useful for many other computer vision tasks (e.g., generic feature extractors) (Hu et al., 2015). The idea of exploring CNN-driven features is also motivated by their usefulness on a wide variety of classification and data-driven modelling tasks. A CNN comprises three categories of various layers including convolutional layers, pooling layers, and fully connected layers. The convolutional layer has a set of learnable filters to convolve through the full input images and deliver various types of activation feature maps. The pooling layer sees the convolutional layer output and reduces the spatial size of the feature maps. The last layer of the CNN are fully connected layers that support all

neurons in the layer and are fully connected to activations in the previous layer and convert the 2D feature maps into 1D feature maps for classification.

The outputs of the CNN layers can be interpreted as the visual features within any image or time-series dataset. CNN models which are trained for classification purposes are used as feature extractors, mainly by removing the output layer. The features extracted from the pre-trained CNN have been successfully used in computer vision tasks such as scene recognition and object characteristic detection and achieve better results compared to handcrafted features. Egede et al. (2017) trained CNN to extract features for pain intensity classification. The same strategy is applied for feature extracting (Haque et al., 2018; Rodriguez et al., 2017; Zhou et al., 2016). This strategy is successfully applied in recognition systems to deal with data defects which is one of the problems in pain recognition (Sharif Razavian et al., 2014).

Many variants of CNN models have achieved increasingly better performance. Examples of these include the renowned ImageNet dataset for object classification such as AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan & Zisserman, 2015), VGGFace (Parkhi et al., 2015), ResNET (He et al., 2016) and GoogLeNet (Szegedy et al., 2015) which is seen to surpass the classification accuracy of human-level performance (Athiwaratkun & Kang, 2015).

AlexNet is the first large-scale CNN model which led to the resurgence of deep neural networks in computer vision (Krizhevsky et al., 2012). The main difference of the AlexNet architect and its predecessors is the increased network depth, which leads to a significantly larger number of tunable parameters, and the use of regularization tricks such as activation dropout and data augmentation (Krizhevsky et al., 2012). VGGNet architect is one of the most popular CNN models. It introduced in 2014 and used in pain detection recently as a pre-trainer. It has two configurations as VGGNet16 and VGGNet19. Like AlexNet, it also uses activation dropouts in the first two fully connected layers to avoid over-fitting (Simonyan & Zisserman, 2015). All previously discussed, pre-trainer networks consist of a sequential architecture with only a single path. GoogleNet consists of a total of 22 weight layers. The basic block of the network is the “Inception Network”. Although the GoogleNet architecture looks much more complex than AlexNet and VGGNet, it involves a significantly reduced number of parameters with better efficiency and higher accuracy performance (Szegedy et al.,

2015). The core idea of ResNet is introducing a so-called "identity shortcut connection" that skips one or more layers. This indicates that the deeper model should not produce a higher training error than its shallower equivalents (He et al., 2016). The VGGFace used CNN implementation based on the VGG-Very-Deep-16 architecture and evaluated on the labelled faces images in the Wild and the YouTube faces dataset (Parkhi et al., 2015). The AlexNet is used as pre-trainer in pain detection systems (Casti et al., 2019). Table 2-2 demonstrates the popular CNN feature extractors and pre-trainers which were recently used in facial expressions as a feature extractor.

Table 2-2. A summary of the key CNN models that have been used in FER.

CNNs	Year	Description
AlexNet (Krizhevsky et al., 2012)	2012	AlexNet comprises 8 layers includes five convolutional layers, some of them followed by max-pooling layers, and the last three are fully connected layers.
VGGNet (Simonyan & Zisserman, 2015)	2015	VGGNet consists of 16 convolutional layers and has only 3x3 convolutions.
VGGFace (Parkhi et al., 2015)	2015	The VGGFace is CNN implementation based on the VGG16 architecture trained by face data.
GoogleNet (Szegedy et al., 2015)	2015	GoogleNet is based on several very small convolutions to drastically reduce the number of parameters. Its architecture consists of a 22 layers deep CNN.
ResNet (He et al., 2016)	2016	It can train 152 layers with lower complexity than VGGNet.

The CNN feature maps can be applied with non-deep learning methods such as Random Forest or a Support Vector Machines (SVM) model to produce data classification results or combine with unsupervised learning techniques such as Principle Component Analysis (PCA), independent component analysis (ICA) and minimum noise fraction (MNF) to select the most important features. PCA

effectiveness in different applications such as facial feature extraction and finding patterns from large dimensional images confirmed specially when it acts as a dimensionality reduction method (Damale & Pathak, 2018; Li et al., 2009). It provides the best set of data dimensions option to improve the model performance, and accelerate the algorithm (Sun et al., 2014). Bargshady et al. (2020a) used a fine-tuned pre-trainer to extract features and then transfer the extracted features into PCA to reduce the dimensions of them. This technique which is used in this research work improve the effectiveness and efficiency of the feature extraction.

2.2 Deep learning Techniques in Image Classification

After extracting the features, the last action of FER is classification of the input face images into one of the pain level categories. In the literature, there are many image classifier systems such as Deep Structured Learning (DSL), *k*-Nearest Neighbor (*k*NN), and SVM. Deep learning classifier systems are recently widely used for image classification purposes. Unlike the traditional methods, where feature extractions and classifications steps are independent, deep learning models can perform it in an end-to-end way. Another option is to employ the CNN as a feature extraction implement and then apply further independent classifiers. The previous results also show that the combination of deep methods with independent classifiers can mean more robust algorithms. Walecki et al. (2017) proposed a hybrid model by merging deep learning with Hidden Markov Model (HMM) where a novel copula-based CNN deep learning approach is used for modelling multivariate ordinal variables. Based on their copula model, which is able to account for the ordinal structures in the output variables and their non-linear dependencies via copula functions modelled as cliques of a Conditional Random Field (CRF), the simulations were jointly optimized with a deep CNN feature encoding layers using a newly introduced balanced batch iterative training algorithm.

Even though CNNs are considerably powerful deep learning techniques for tasks estimation, however; they are not influential in developing sequential data such as video data analysis. Recurrent Neural Networks (RNNs) designed to represent features in capturing information from all the earlier time steps and to renew its representation through upcoming information (Zhou et al., 2016). In a pain detection task, the study of (Martinez et al., 2017) used the LSTM-RNN algorithm to detect pain from the facial

dataset. This approach used the key algorithm to automatically estimate PSPI levels from face images. The expected scores were fed into the customized Hidden CRFs (HCRFs) to estimate the VAS for each person (Martinez et al., 2017; Soar et al., 2018). On the other hand, Bargshady et al. (2020a) developed a different technique which contains a fine-tuned VGGFace pre-trainer joined to a PCA to reduce the dimensionality of the extracted features and then when applied to another classifier, as a joint CNN-LSTM model, consists of a two stream hybrid deep learning technique. Wang and Sun (2018) combined a deep learning and a hand-crafted method by using a deep 3-dimensional convolutional network from video frames as input to extract spatiotemporal facial features and hand-crafted features to extract the geometric information. Egede et al. (2019), used a three streams network using three different feature extraction techniques including appearance HOG, CNN, and shape features using handcrafted algorithms and Relevance Vector Machine (RVM) for pain estimation. In a different way, Chen et al. (2019) proposed an automated pain detection system including two machine learning systems: an Automated Facial Expression Recognition (AFER) system that computes frame-level confidence scores for single AUs and a Multiple Instant Learning (Milgram et al.) system that performs sequence-level pain prediction based on contributions from a pain-relevant set of AU combinations. More details about automatic pain recognition approaches are explained in survey paper published recently (Werner et al., 2019).

Table 2-3 shows the summary of the literature which applied deep learning in pain detection from facial expressions and Table 2-4 indicated the-state-of-the-art non deep leaning techniques applied for the same task.

Table 2-3. A summary of literature that used deep learning models to detect pain from facial expressions.

Paper	Feature	Classifier	Metric	Score (%)	Database
(Rodriguez et al., 2017)	VGGFace	LSTM	AUC, MAE, MSE, PCC, ICC	93.3, 0.5, 0.74, 0.78, 0.45	UNBC-McMaster
(Martinez et al., 2017)	AAM	BiLSTM + HCRF	MAE, ICC	0.94, 0.30	UNBC-McMaster
(Haque et al., 2018)	VGGFace	LSTM	Mean Frame, Mean Sequence	18.17, 18.55	Multimodal Intensity Pain (MIntPAIN)
(Walecki et al., 2017)	VGG16	CRF	MAE, ICC	0.61, 0.45 and 1.23, 0.63	UNBC-McMaster
(Xu et al., 2015)	CNN MSRA-CFW	CNN.	Mean accuracy	81.5	PICS
(Egede et al., 2017)	ASM	CNN + RVR	RMSE, CORR	0.99, 0.67	UNBC-McMaster
(Zhou et al., 2016)	AAM	RCNN	MSE, PCC	1.54, 0.65	UNBC-McMaster
(Bellantonio et al., 2016)	VGGFace	CNN + LSTM.	F measure	0.69	UNBC-McMaster
(Bargshady et al., 2020a)	VGGFace + PCA	CNN+LSTM	Accuracy, AUC	91.2, 98.4	UNBC-McMaster
(Wang & Sun, 2018)	HOG	CNN + SVR	RMSE, PCC	0.94, 0.68	UNBC-McMaster
(Wang et al., 2017)	HOG	CNN	MAE, MSE	0.991, 1.720	UNBC-McMaster
(Tavakolian & Hadid, 2019)	ResNet	ResNet	MSE, AUC	0.32, 98.53	UNBC-McMaster and BioVid
(Salekin et al., 2019)	VGG16	LSTM	Accuracy, AUC	92.48, 90	Infant COPE
(Theagarajan et al., 2018)	CNNs	LSTM	accuracy, precision, recall	94.85, 94.86, 96.29	Infant COPE
(Kharghanian et al., 2016)	CDBN	SVM	Accuracy, F-measure, AUC	87.2,86.44, 94.48	UNBC-McMaster

Table 2-4. A summary of the state-of-the-art literature that used non deep learning models to detect pain from facial expressions.

Paper	Feature	Classifier	Metric	Score (%)	Database
(Lucey, Cohn, Prkachin, et al., 2011)	AAM	SVM	AUC	83.9	UNBC-McMaster
(Lucey, Cohn, Matthews, et al., 2011)	AAM	SVM	AUC	84.7	UNBC-McMaster
(Lucey, Cohn, Lucey, Matthews, et al., 2009)	AAM	SVM	AUC	78	UNBC-McMaster
(Lucey, Cohn, Lucey, Sridharan, et al., 2009)	AAM/ASM	SVM	AUC	78.4	UNBC-McMaster
(Kaltwang et al., 2012)	AAM, LBP	RVR	MSE, PCC, ICC	1.39, 0.59, 0.50	UNBC-McMaster
(Zhao et al., 2016)	LBP, Gabor	OSVR	MAE, PCC, ICC	0.81; 0.60; 0.56	UNBC-McMaster
(Florea et al., 2014)	HOT	SVR	MSE, PCC	1.21, 0.53	UNBC-McMaster
(Ashraf et al., 2009)	AAM	SVM	Hit rate	82	UNBC-McMaster
(Hammal & Cohn, 2012)	AAM	SVM	Recall, F1	61, 57	UNBC-McMaster
(Rudovic et al., 2013)	LBP	KCORF	Precision, F1	65, 40.2	UNBC-McMaster
(Khan et al., 2013)	HOG-LBP	SVM, RF	Accuracy	96.4	UNBC-McMaster
(Pedersen, 2015)	Custom Features	SVM	AUC, Accuracy	96.5, 86.1	UNBC-McMaster
(Rathee & Ganotra, 2015)	feature deformation	SVM	Accuracy	96.0	UNBC-McMaster
(Yang et al., 2016)	LBP	SVM	Accuracy	83.4, 71	UNBC-McMaster and BioVid

2.3 Pain Databases from Facial Expression

Unfortunately, for various justifiable reasons (*e.g.*, ethical considerations, human data confidentiality or institutional regulations) there appear to be not enough publicly available medical databases that were freely available to any researcher in the area of facial expression analysis for pain detection. Most of the high-quality databases could potentially help design a robust facial image recognition system often requires permission prior to their accessibility. The authorization for the usage of such databases is generally feasible since most of these databases only require a user-defined form to be completed as a formality to access the records. Table 2-5 shows the databases denoting facial pain intensity with their relevant accessibility and other details..

Table 2-5. Databases of facial expressions related to pain.

Databases	Samples	Subjects	Pain Type	Description
UNBC-McMaster Shoulder Pain Expression (Lucey, Cohn, Prkachin, et al., 2011)	200 video sequences, 48398 images	25	Self-identified pain patient, Natural Shoulder pain	Includes FACS, 66-point landmarks, and PSPI codes for 16 pain level.
BioVid (Walter et al., 2013)	17300 videos	90	Healthy volunteers, Stimulated heat pain	4 pain (Stimuli) level
MIntPAIN (Haque et al., 2018)	9366 video sequences, 187939 images	20	Healthy volunteers, Stimulated electrical pain	5 pain (Stimuli) level
Infant COPE (Brahnam et al., 2007)	204 facial images	26	Healthy, heel lancing pain simulation	pain, crying, heel friction, nasal air, stimulus, rest
Hi4D-ADSIP (Matuszewski et al., 2012)	240 pain sequence	80	Healthy	Pain and emotion
BP4D (Zhang et al., 2014)	41 2D, 41 3D pain video	41	cold-pressor test	Pain and emotion
EmoPain (Aung et al., 2015)	44 video, 50071 pain frame	48	Chronic back pain	Pain from face and body movement

One particularly useful database for a facial image and pain recognition system is the UNBC-McMaster Shoulder Pain Archive (Lucey, Cohn, Prkachin, et al., 2011). This database has been pre-processed and includes noise-free backgrounds, consists of carefully labelled images with the required facial expressions. The UNBC-McMaster Shoulder Pain Archive provides videos with each frame that is coded in terms of PSPI score (Prkachin & Solomon, 2008), and is defined on an ordinal scale 0-15. These data were collected by researchers at two major institutions: the McMaster University and the University of Northern British Columbia, under a research program devoted to a better understanding of the properties of facial expressions of pain. The process of data collection involved the identification of pain expressions and the role of pain expressions in clinical assessment of people suffering from such conditions. Here, the participants provided informed consent for the use of their facial images for scientific

studies on the perception of their pain, including the pain detection. The frames of these images were labelled using the validated PSPI score based on the FACS. Specifically, these aimed to code the different movements of the facial muscles to explain the different pain intensity levels. To collate these images, spontaneous expressions of the corresponding pain from these patients were recorded using a digital camera in a laboratory whilst they underwent eight standard range-of-motion tests. Statistically the database includes a total of 200 sequences across 25 different subjects, totaling 48398 images (Lucey, Cohn, Prkachin, et al., 2011).

Another useful database is the BioVid repository that includes 17300 videos from a total of 90 subjects that can be used to estimate the pain from their facial images. To compensate for the varying heat pain sensitivities among these participants, the stimulation temperatures were therefore adjusted based on the subject-specific pain threshold and pain tolerance. BioVid data contains five sections and section B consists of pain stimulations with facial EMG sensors (Walter et al., 2013), providing a significant number of features that may be extracted to develop an automated pain recognition system.

The MIntPAIN has been generated recently. The MIntPAIN repository has multimodal pain-related data that were obtained by providing electrical stimulations in five different levels for a total of 20 healthy subjects. Each subject completed two trials during the data capturing session and each of those trials had 40 sweeps of pain stimulations. In each sweep, data were captured in two distinct parts: one for ‘no pain’ and the other for ‘one of the four pain levels’, although some sweeps were missed for only a few subjects (Haque et al., 2018).

Infant COPE has been collected from infant facial expression. Infant COPE includes a total of 204 color photographs were taken of 26 Caucasian neonates (13 boys and 13 girls) ranging in age from 18 h to 3 days old. All infants were in good health. The facial photographs consist 67 in resting mood, 18 in crying, 23 in air stimulus, 36 in friction and 60 in acute pain (Zhang et al., 2014).

Hi4D-ADSIP database contains 3360 3-D dynamic high-resolution sequences from 80 subjects of seven expression categories: anger, disgust, fear, happiness, sadness, surprise, and pain. The database consists of 48 female and 32 male subjects from a variety of ethnic origins. The database has been validated using psychophysical

experiments used to formally evaluate the accuracy of the recorded expressions (Matuszewski et al., 2012).

BP4D 3D includes video database of spontaneous facial expressions in a diverse group of young adults (23 women, 18 men) Frame-level ground-truth for facial actions has been obtained using the FACS. Facial features track in both 2D and 3D domains. Cold pressor uses by submerging a hand in ice water for as long as possible to provide physical pain (Zhang et al., 2014).

EmoPain includes body movement and facial expression videos from potential participants were identified by health care staff from the Pain Management Centre at the National Hospital for Neurology and Neurosurgery, United Kingdom. The dataset will be made available to the research community via a web-accessible interface linked. The first release will contain eight continuous facial pain ratings and the temporal annotations for movement-based pain behaviors from four raters, and the approximate onset and end timings of each exercise will also be provided for the patient set. One challenge in the using the EmoPain for facial expression is the original images is not accessible and only extracted features from specific feature extractor were available (Aung et al., 2015).

2.4 Chapter Summary

In this chapter the related literature about feature extraction methods were discussed. The traditional and non-deep learning techniques and deep learning feature extraction methods were explained. Both deep learning and non-deep learning classifiers applied to pain detection from facial expressions were reviewed and their strength and weaknesses, number of recognized classes, the applied databases, measurement metrics were elaborated and compared. Then the popular and available databases in pain detection from facial expressions to train and evaluate the models were discussed. The previous studies review show that pain recognition algorithms based on deep learning models to detect pain levels from facial expression in multi classes still need improvement. In this research new enhanced deep learning models for this task were developed and evaluated and in the following sections details about the proposed methods, results, and comparison with the state-of-the-art were discussed.

CHAPTER 3

Research Methodology

In this chapter the applied research methodology and the proposed research design and approaches applied in this thesis research work is introduced and explained. The research design framework and its components including doing literature review, developing conceptual framework, developing theoretical framework, selecting databases, doing experimental, evaluation the developed frameworks, and writing and documentation are described. Several research approaches appeared to be legitimate within the field of knowledge discovery and information systems. These methods include case study, field studies, action research, prototyping, and scientific such as experimental methods. As this research focuses on the development of robust mechanisms in the knowledge discovery system, these mechanisms or proposed theories must be proven by the classic scientific method of experiment. This research needs diagnosis of the problem and development the solution for the problem as identified in action research. The scientific experimental approach integrated with action research is chosen as the research method. The main features of each approach (scientific method and action research) are outlined and justified, below.

3.1 Scientific Approaches

Scientific approaches may be defined as those that have arisen from the scientific tradition – characterized by repeatability, reductionism and refutability – and which assume that observations of the phenomena under investigation can be made objectively and rigorously (Lyytinen & Klein, 1985). The scientific method is common to many disciplines such as biology or sociology and only the tools of research are different (Leedy & Ormrod, 2005).

3.2 Action Research Approach

General action research (AR) is viewed as a cyclical process (Susman, 1983). This process contains four major phases: plan, act, observe and reflect. It aims to link theory and practice, achieving both practical and research objectives. In this research, action

research is used as one of the primary research approaches because: (1) it provides a general guideline and methodology to perform research activities in a logically and efficient way; (2) action research has the strength of evaluating and reflecting on learning (Susman, 1983). Evaluative and reflective learning enables the researcher to step back and critically analyze an action, decision, or product by focusing on what is done or being done that incorporates learning to be applied to a new situation (Susman, 1983). As this thesis research project focuses on the development of the knowledge discovery system, evaluation and reflection were needed to find the best solution; scientific methodology integrated with action method is applied.

3.3 Research Design and Approach

A research design is the arrangement of conditions for the collection and analysis of data in a manner that aims to combine relevance to the research purpose with economy in a procedure (Selltiz et al., 1976). It is a blueprint for the collection, measurement, and analysis of data (Phillips, 1966). A research methodology is a way in which one proceeds to solve the problem and a description of how the research will be conducted (Leedy & Ormrod, 2005). It is an operational plan generated from a research design. It also provides a more detailed description of the approach taken in carrying out the research, such as the characteristics of data, data collection instruments, and the data collection process. For example, description of the sample size and the origin of data, description of data collection instruments, and pretext of these instruments (Gable, 1994). The research design conducted in this doctoral thesis can be described as (1) exploratory, (2) observational, (3) experimental, and (4) descriptive. The proposed research methodology generated from the research design combines the strength of both the scientific (empirical) approach and action research to achieve the research objectives. It includes seven main phases. (1) literature review; (2) constructing a conceptual framework; (3) developing theoretical models; (4) data selection; (5) experimental configuration including building a prototype system, and carrying out several experiments; (6) undertaking laboratory evaluation and reflection; (7) interpreting and analyzing results, and thesis writing. Fig. 3-1 shows the research design schema.

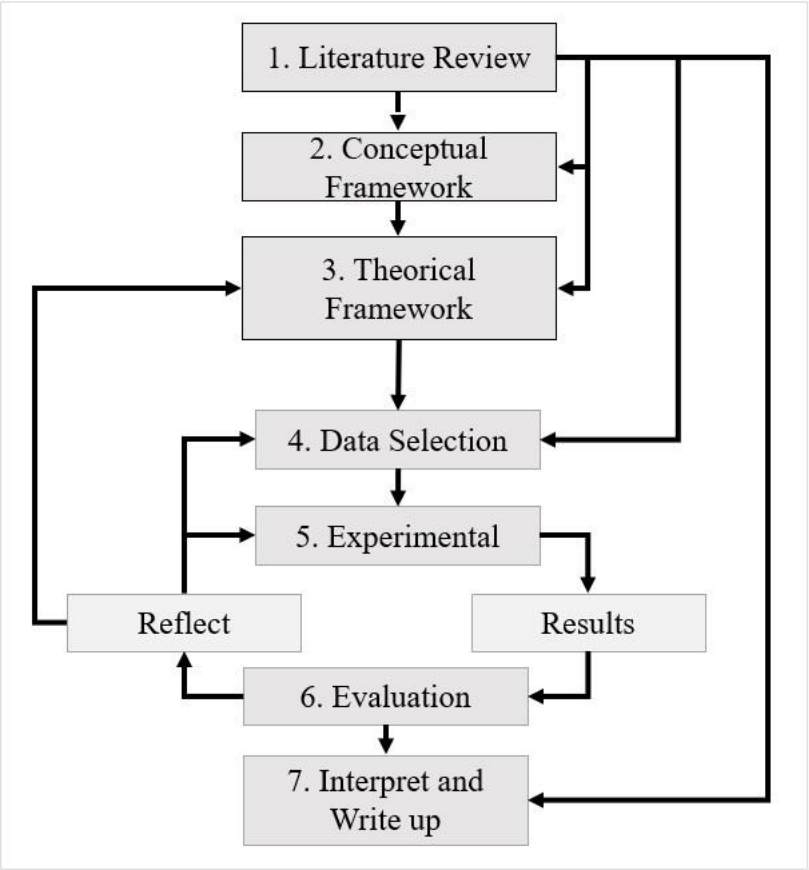

Fig. 3-1. Research design framework developed in this doctoral thesis.

3.3.1 Literature Review

A literature review involves the researcher exploring the literature to establish the status quo, formulating a problem or research inquiry, defending the value of pursuing the line of inquiry established, and comparing the findings and ideas with his or her own (Bruce, 1994). This involves the synthesis of the work of others in a form that demonstrates the accomplishment of the exploratory process. This phase explores potential important issues/problems, relationships and relevant theories identified from past research, and focuses on the emerging fields for proposed research. The major tasks were critical analysis and evaluation of the literature to answer research question one. The several crucial related topics: automated pain recognition from facial expression, image processing, deep learning feature extraction, deep learning classification, related and available pain database from facial images were discussed in the literature review.

3.3.2 Conceptual Framework

The general image processing conceptual framework strategy for facial expression are described in this section which shows the roadmap for this research. After deciding on the problem, a preliminary literature review is used to define the conceptual framework to conduct an automated pain detection system from facial expressions based on image processing techniques. Based on the schematic view of the general FER framework used in a practical healthcare environment tool (Chibelushi & Bourel, 2003), the conceptual framework is designed as shown in Fig. 3-2.

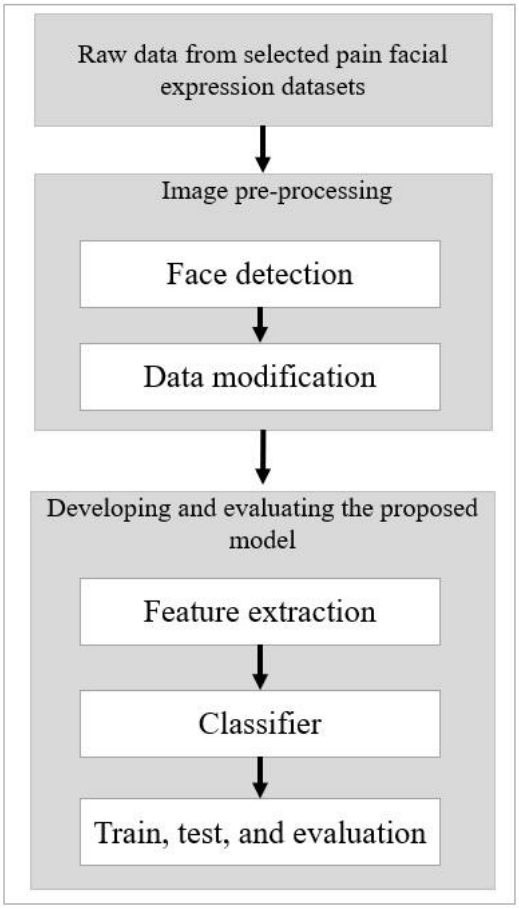

Fig. 3-2. Proposed conceptual framework

3.3.3 Theoretical Framework

New deep learning models for extracting features and detecting pain intensity from facial expression video images were developed to answer research questions two and three. The developed theorical models for feature extraction and pain intensity classification and their results are described in Chapter 4, 5, and 6. The purpose of this

is to develop and design a new mechanism based on the literature review to solve problems of pain recognition automatically from facial expressions' images and estimation of pain intensity levels.

3.3.4 Data Selection

To train and test the proposed deep learning model to recognise pain from facial expression, two popular databases the UNBC-McMaster Shoulder Pain Archive database (Lucey, Cohn, Prkachin, et al., 2011) and MIntPAIN (Haque et al., 2018) were selected. Unfortunately, the number of databases for pain detection from facial expression is very limited and, in many cases, such as some of the databases described in section 2.5 of the previous chapter, accessing the patients' raw image data is restricted. These two databases were selected since their images and labels were available to use and contain very useful information for pain analysis and can be used in any algorithm. In the following sections detailed information for both databases, their data characteristics and the selected datasets used for the experiment are described.

3.3.4.1 UNBC-McMaster Shoulder Pain Archive Dataset

The UNBC-McMaster Shoulder Pain Archive database (Lucey, Cohn, Prkachin, et al., 2011) provides video frames within a set of video sequences with each frame labeled in terms of the PSPI score. This database also provides a total of 200 sequences across 25 subjects with a total of 48,398 facial images. Fig. 3-3 shows some of these images indicated by the PSPI.

Fig. 3-3. Image frame samples of the UNBC-McMaster Shoulder Pain Archive database (Lucey, Cohn, Prkachin, et al., 2011) used in this study

Like many image-based datasets, the database is unbalanced, and it is very challenging to perform the modelling experiments. This meant that, as shown in Fig. 3-4, the number of no pain images PSPI score labels were higher than other labels and the number of images with PSPI labels greater than 4 is few in this database. Based on the specific character of the database it is likely that any model would be biased towards the prediction of no-pain at the cost of missing pain frames.

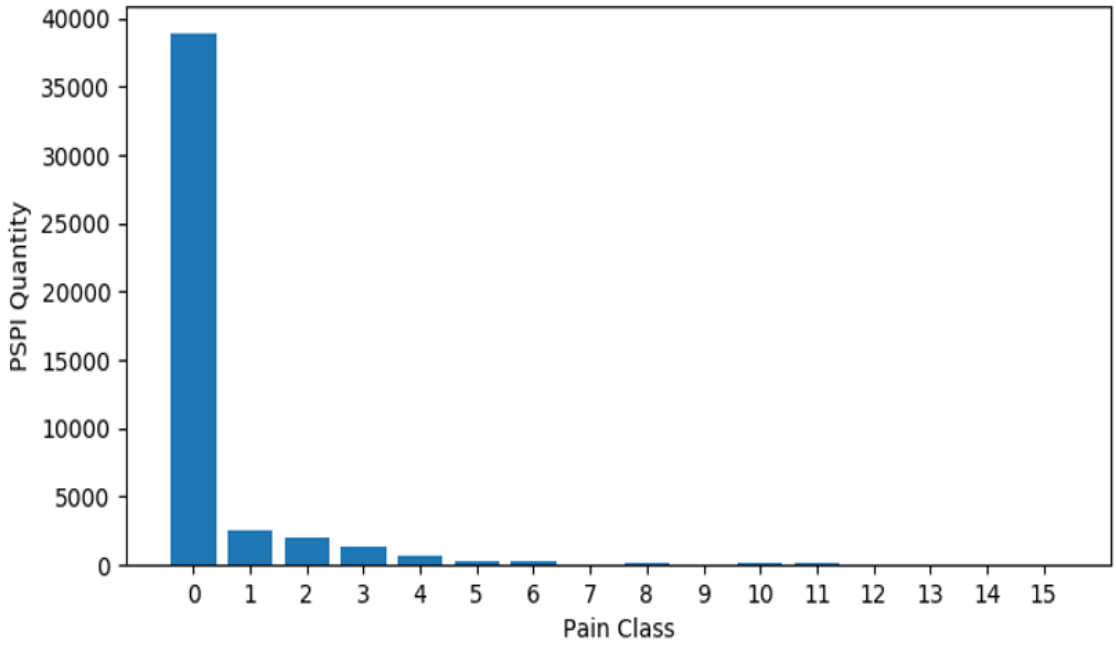

Fig. 3-4. Amount of the PSPI code per each class in the UNBC McMaster Shoulder Pain Database

It must be noted that using imbalanced data basically means the researcher is intentionally biasing the data to potentially get an interesting result. To deal with this issue, in this thesis, a selected dataset of the database is applied. To doing this, the main database is balanced using under resampling techniques to reduce the majority class (no-pain class). Full sequences included only no pain (PSPI = 0) frames were removed and some no-pain frames from the beginning and end of sequences, which included no-pain frames, were removed. 10,783 images were used in this research. To create the training sequence for the proposed video analyses algorithm, the frames were firstly sorted out in time domain, and subsequently, each sequence is set to 20 frames. On the other hand, the number of classes for more than PSPI = 4 is few. the classes with PSPI = 4 and greater than 4 were categorized as a strong pain level. In order to balance data for each class, the database is divided as follows, no-pain (PSPI = 0), weak-pain (PSPI = 1), mid-pain (PSPI = 2 and 3), and strong-pain (PSPI > = 4). As described in Table 3-1, the selected dataset from the UNBC-McMaster shoulder Pain database has four classes.

Table 3-1. Divided levels of pain in the database for four levels based on PSPI codes of images' frames.

PSPI Score	Pain Level	Number of images
0	No pain	2483
1	Weak pain	2871
2 and 3	Mid pain	3757
4 and >4	Strong pain	1672

3.3.4.2 MIntPAIN Dataset

The MIntPAIN database includes pain video data taken by electrical stimulation in five levels (Level 0 – no pain to Level 4 – highest pain level) of 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 the 20 subjects has been selected. The number of no pain video sequences were more than any other. based on the specific character of the database it is likely that any model would 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 has been balanced using under resampling techniques to reduce the majority class (no-pain class). So, some no pain sequences were removed.

The resampling technique is applied on the selected dataset since a few subjects were missing for some sweeps and there is also not an equal proportion for each class. the under-sampling technique is applied to reduce the majority class, and some no painful sequences (label 0) were removed. The total of 34800 video frames is selected for experimentation in this research. Fig. 3-5 shows the samples of the selected dataset.

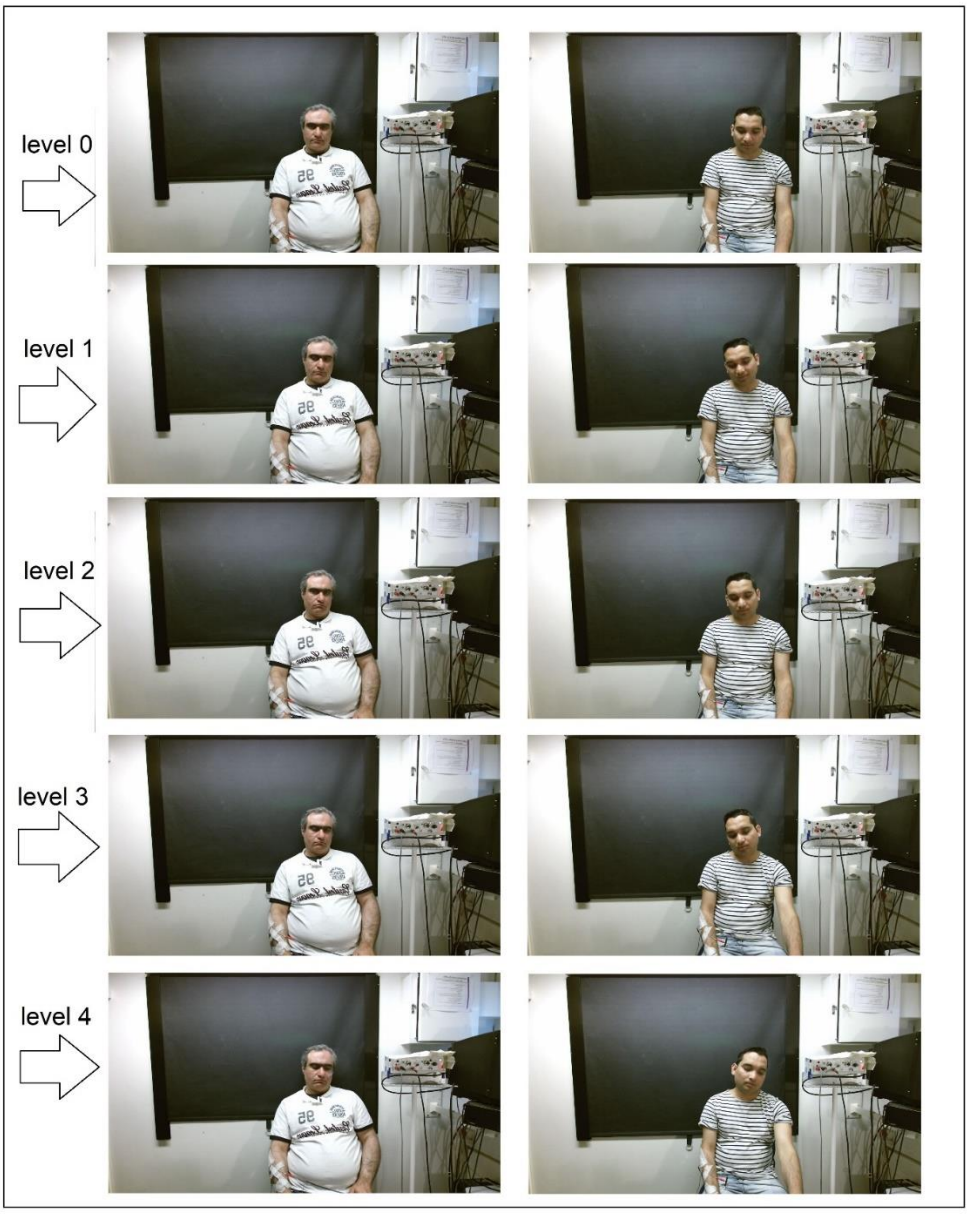

Fig. 3-5. Samples of selected dataset of MIntPAIN database (Bellantonio et al., 2016; Haque et al., 2018).

3.3.5 Experimental Configuration and Results

A prototype modelling system is developed to train, test, and evaluate the proposed enhanced deep learning pain detection model to answer research questions four and five. Modelling experiments were conducted to validate the effectiveness and efficiency of the proposed models in chapters 4, 5, and 6. The developed algorithms were executed under *Intel Core i7 @ 3.3 GHz* and *16 GB* memory computer. *Python* software (Sanner, 1999) is used for the model construction and prototyping, since it has freely available library suits for deep learning such as *Keras* (Ketkar, 2017), *TensorFlow* (Abadi et al., 2016), *Scikit-learn* (Pedregosa et al., 2011) and *Matplotlib* (Hunter, 2007). *Keras* allows for easy and fast prototyping and supports both convolutional networks and recurrent networks. *Matplotlib* is a *Python 2D* plotting library that is used for plotting and statistical analysis of modelling data.

3.3.6 Evaluation and Reflection

Evaluation of the obtained results is one of the most important parts of the research to estimate the effectiveness and efficiency of the proposed models and answer research questions 4 and 5. To evaluate the proposed algorithms, training and testing datasets were divided by *k-fold cross validation* by *k=10*. Cross-validation is a computer intensive technique, using all available examples as training and test examples. It mimics the use of training and test sets by repeatedly training the algorithm *K* times with a fraction *1/K* of training examples left out for testing purposes.

In practice, the data set *D* is first chunked into *K* disjoint subsets (or blocks) of the same size $m \triangleq n/K$.

If T_k for the K_{th} such block, and $D_k$the training set obtained by removing the elements in T_k from D.

The cross-validation estimator is defined as the average of the errors on test block T_k obtained when the training set is derived from T_k (Bengio & Grandvalet, 2004):

$$CV(D) = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{m}\sum_{z_i \in T_k} L(A(D_k), z_i). \quad (3\text{-}1)$$

To evaluate the generality of the proposed algorithms the Leave One-subject Out Cross Validation (LOOCV) is also applied.

If data

$y_1, \dots y_n$

modelled as independent given parameter θ; thus,

$$p(y|\theta) = \prod_{i=1}^{n} p(y_i|\theta). \quad (3\text{-}2)$$

This formulation also encompasses latent variable models with

$p(y_i|f_i, \theta)$,

where

f_i

are latent variables. Suppose of a prior distribution

$p(\theta)$,

generate a posterior distribution

$p(\theta|y)$

and a posterior predictive distribution

$$p(\tilde{y}|y) = \int p(\tilde{y}_i|\theta)p(\theta|y)d\theta. \quad (3\text{-}3)$$

To maintain comparability with the given dataset and to obtain easier interpretation of the differences in scale of effective number of parameters, a measure of predictive accuracy for the n data points taken one at a time defines as:

$$\text{elpd} == \text{expected log pointwise predictive densityfor a new dataset} = \sum_{i=1}^{n} \int p_t(\tilde{y}_i) log p(\tilde{y}_i|y) d\tilde{y}_i \quad (3\text{-}4)$$

where

$p_t(\tilde{y}_i)$

is the distribution representing the true data-generating process for $\tilde{y}_i$.

The LOOCV estimate of out-of-sample predictive fit is

$$elpd_{loocv} = \sum_{i=1}^{n} \log p(y_i|y_{i-1}) \quad (3\text{-}5)$$

where

$$p(y_i|y_{i-1}) = \int p(y_i|\theta)p(\theta|y_{i-1})d\theta \quad (3\text{-}6)$$

is the leave-one-out predictive density given the data without the i_{th} data point (Vehtari et al., 2017).

Several performance evaluations measures, including classification accuracy, Mean Absolute Error (MAE), Mean Squared Error (MSE), Area under Curve (AUC), Precision-recall curves (PR curves), and F-measure were used to evaluate the performance of the proposed model.

Classification accuracy is the ratios of the number of correct predictions to the total number of input samples in both training and testing sets.

MAE is the average of the difference between the original values and the predicted values. It gives us the measure of how far the predictions were from the actual output. However, they do not give any idea of the direction of the error whether the data is under predicting or over predicting.

MSE is quite like MAE, and the only difference is that MSE takes the average of the square of the difference between the original values and the predicted values. The advantage of MSE is that it is easier to compute the gradient, whereas MAE requires complicated linear programming tools to compute the gradient. Mathematically, the metrics were stated as follows where:

$$e = e_{experimental} - e_{true} \tag{3-7}$$

and

N = number of errors:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|e_i| \tag{3-8}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 \tag{3-9}$$

True Positive Rate (TPR) corresponds to the proportion of positive data points that were correctly considered as positive, with respect to all positive data points. False Positive Rate (FPR) corresponds to the proportion of negative data points that were mistakenly considered as positive, with respect to all negative data points. FPR and TPR both have values in the range [0, 1]. The algorithms calculate TPR and FPR metrics and applied them for the measuring of the AUC and F measures.

AUC is one of the most widely used metrics for evaluation. AUC of a classifier is equal to the probability that the classifier will rank a randomly chosen positive example higher than a randomly chosen negative example. AUC is the area under the curve of the plot FPR vs TPR at different points in [0, 1] (Powers, 2011). The area under the receiver operating characteristic (Bellantonio et al., 2016) curve (i.e., the AUC) is used as a performance measure for these machine learning algorithms following other works e.g., (Bradley, 1997). The details of the AUC calculation were described in Bradley (1997).

The F-measure is used to measure a test's accuracy, and it balances the use of precision and recalls doing it. The F measure can provide a more realistic measure of a test's performance by using both precision and recall. F-measure and precision were calculated based on False Positive (FP), True Negative (TN), False Negative (FN), and True Positive (TP). PR curves is based on precision rather than the false positive rate, and better reflect model performance when predicting rare outcomes (Davis & Goadrich, 2006). the PR curve contains TP/(TP+FN) on the y-axis and TP/(TP+FP) on the x-axis. It is a curve that combines precision and recall in a single visualization. More information and formula about measuring PR curves could find in (Boyd et al., 2013). In this thesis experimental the *sklearn Python library* has been used to calculate PR curves.

In the following the equation of some metrics based on TP and FN were described:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3-10}$$

$$Precision = \frac{TP}{TP+FP} \tag{3-11}$$

$$Recall = \frac{TP}{TP+FN} \tag{3-12}$$

$$F1 = 2 \times \frac{Recall \times Precision}{Recall+Precision} \tag{3-13}$$

3.3.7 Interpretation and Write up

The overall results produced by developing the model, the results of experimental, and literature review were analyzed, interpreted, and reported. The research results were presented in appropriate publications.

3.4 Chapter Summary

In this chapter the research methodology concepts and types applied in this doctoral thesis is discussed. The research design framework, steps, and its components development to answer the related research questions is explained. The applied databases, configuration set ups, evaluation metrics which is utilized in the proposed models is introduced and clarified. The goal of this chapter is to provide an overview for the content of Chapters 4, 5 and 6 which were elaborated in the following chapters. In the following chapters three proposed models to pain recognition system from facial expression video images is described and discussed.

CHAPTER 4

The Proposed Hybrid CNN-BiLSTM (EJH-CNN-BiLSTM) Model

This Chapter reports on the successful design of an enhanced joint hybrid deep learning approach, utilizing CNN linked to a joint bidirectional long short term (BiLSTM) neural network algorithm to address existing challenges of pain intensity estimation. The study applies the UNBC-McMaster Shoulder Pain Achieve database (Lucey, Cohn, Prkachin, et al., 2011), to train the proposed algorithm and subsequently, detect the pain in four different levels efficiently and effectively. To attain this, the popular VGGFace pre-trainer (Parkhi et al., 2015) was fine-tuned and utilized to extract the features from the facial image dataset. To improve the computational efficiency of the algorithm, the full dataset was reduced to only the most significant input features by applying the PCA, and the outputs of the selected features were then transferred to the CNN-BiLSTM joint network for the classification of pain intensity. The novelty in the study lies in developing a new EJH-CNN-BiLSTM algorithm that is able to extract and select most prominent features and classify pain levels. The newly proposed system was then conditioned in such a way that the outputs of the machine learning algorithm were further tuned to estimate four pain levels ranging from 0 to 3 [0 = no pain and 3 = strong level of pain]. The contributions of this research work will result in:

1. An efficient and effective algorithm, based on transfer learning, that is designed to extract the most important features by using a fine-tuned VGGFace and the PCA dimension reduction method.
2. A new enhanced joint hybrid deep learning approach that automatically estimates four-levels of pain from facial expressions is proposed.
3. The newly proposed EJH-CNN-BiLSTM model consists of both the feature extraction and the classification algorithm in a single workflow that can be easily implemented in a real-time online system.

A block-diagram of the proposed modelling framework established in this study is illustrated in Fig 4-1. Basically, it is divided into three primary components that aim to improve the overall efficacy of the algorithm. In the first step, the original images (captured as video frames) were transferred to the pre-processing stage, which applied

procedures related to the cropping, resizing and normalizing techniques required to adjust the original images before being incorporated into the feature extraction phase and the model training stage. Subsequently, in a new proposed framework for feature extraction and selection, the fine-tuned pre-trained CNN framework was applied to extract these features. This was later output into the PCA stage, aimed to reduce the dimensionality of the extracted features from the video's images. Since this study used video-image based data, additional temporal information is necessary to improve the classification model. It should be noted that the BiLSTM algorithm, used as a specific type of RNN neural net, is especially suited for sequential datasets since their neurons do not only have connections between the next layers but also have connections to themselves, which aim to capture the past and future input features. the extracted features in a sequence length were transferred into a newly developed Enhanced Joint Hybrid classifier algorithm, denoted in this study as the EJH-CNN-BiLSTM in order to obtain four distinct pain intensity levels. The details of the proposed model were explained in the following subsection.

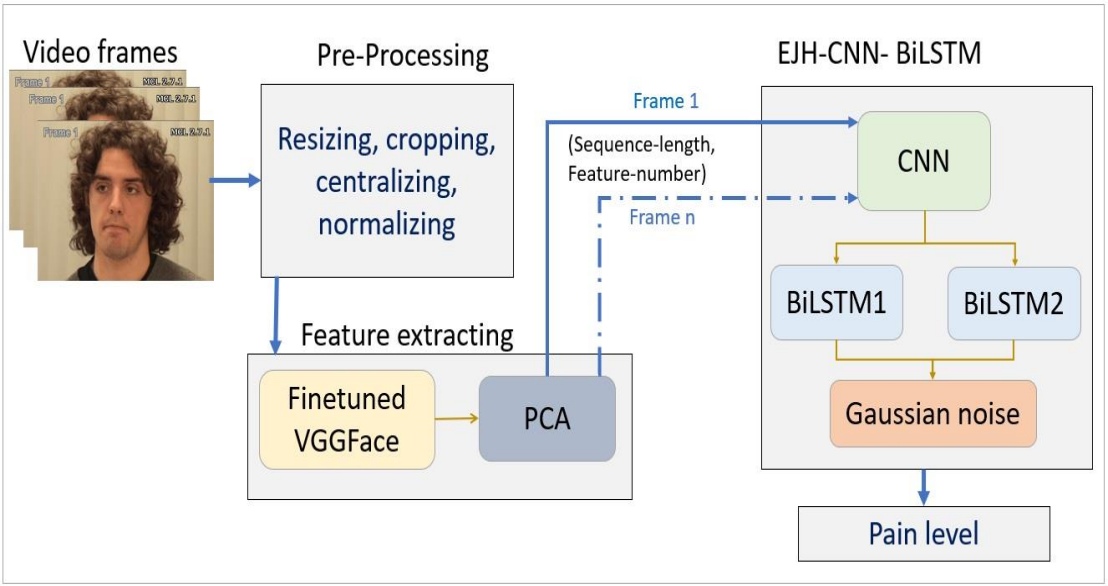

Fig. 4-1. The proposed EJH-CNN-BiLSTM model designed for pain detection from facial expression images.

4.1 Image Preprocessing

For a better performance of the proposed algorithm, applying image pre-processing for the raw selected datasets is essential. For this purpose, some important image pre-processing techniques including face detection, image cropping, centralizing, and

resizing were applied to the raw images. Face detection and cropping is also an important task to achieve high recognition rate. Face detection involves detecting a face from an image using complete image (image-based approach) or by detecting one or more features from the image (Feature based approach) such as nose, eyes, lips etc. Face detection can also be done based on active shape models such as locating head boundary. The *OpenCV* face detection library (Emami & Suciu, 2012) and *detectMultiScale ()* method (Howse et al., 2016) was applied to detect face from the raw image. Algorithm 4-1 shows the related procedure.

Algorithm 4-1: Face detection algorithm using OpenCV and detectMultiScale ()

```
1   Procedure FaceDetection (image)
2       import cv2
3       import numpy as np
4       face_casecade = cv2.CascadeClassifier (image)
5       img = cv2.imread (imgfile_path)
6       img_copy = np.copy (img)
7       faces = face_cascade.detectMultiScale (img, 1.25, 6)
8       for f in faces:
9           x, y, w, h = [v for v in f]
10          img = cv2.rectangle (img_copy, (x,y), (x+w, y+h), (255,0,0) , 3)
11          img = img [y:y+h, x:x+w]
12      end for
13  end Procedure
```

In a real-world scenario, an image dataset may be taken in a variety of conditions such as different orientations, location, scales, and brightness. the conventional image pre-processing technique such as the illumination, normalization, cropping, and centralizing (Fig. 4-2) were applied in these raw images to improve the identification of the images during any experimental phase. Cropping was done using the face detection algorithm. The images centralizing applied on the images as a pre-processing technique.

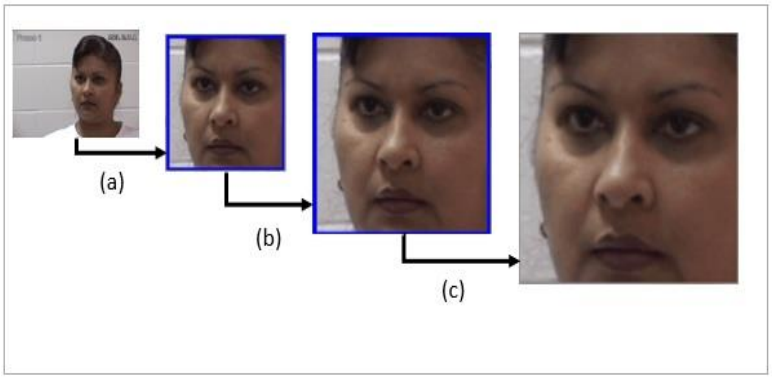

Fig. 4-2. Image pre-processing steps for a sample image data (a) face detection, (b) centralizing, (c) resizing.

Algorithm 4-2 shows the process of the images' centralizing. *Numpy zeros* as *np.zeros()* function in *python was* used to get an array of given shape and type filled with zeros**.** Three parameters can pass inside function *np.zeros* were shape, *dtype* as *np.zeros (shape, dtype, order)* (McKinney, 2012). The *numpy.unit8* as *np.unit8* describes unsigned integer (0 to 255). Finally, the input images were resized to *224×224×3* pixels because this representation is the most common input size for most of the deep neural network models.

Algorithm 4-2: Images' centralizing process

```
1   Procedure centering_image (img_shape[])
2       import numpy as np
3       size = [256,256]
4       img_size = img_shape [:2]
5       row = (size [1] - img_size [0]) // 2
6       col = (size [0] - img_size [1]) // 2
7       resized = np.zeros (list (size) + [img.shape[2]]) , dtype = np.unit8)
8       resized [row: (row + img.shape [0]), col: (col + img.shape[1])] = img
9       return resized
10  end Procedure
```

To normalize the pixel values for both the training and testing datasets, these data were rescaled to the range of [0,1]. This involved first converting the data type from unsigned integer to float values, and then dividing the pixel values by the maximum value (Schertler, 2014).

$$Normalize: R \to R: x \to \frac{x}{d} \qquad d = \max_{x \in image} \| x \| \tag{4-1}$$

4.2 Proposed Feature Extraction Model

Feature extraction and representation is a crucial step for multimedia processing. How to extract ideal features that can reflect the intrinsic content of the images as complete as possible is still a challenging problem in computer vision. How to find effective features is the core issue in image classification and pattern recognition. Humans have an amazing skill in extracting meaningful features, and a lot of research projects were undertaken to build a facial expression system as smart as human in the last several decades.

Deep learning has been widely applied to several real-world applications. However, most existing supervised algorithms work well only under a circumstance such as the training and test data should be represented by the same features and drawn from the same distribution. the performance of these algorithms heavily depends on collecting high quality and sufficient labeled training data to train a statistical or computational model to make predictions on the future data. However, in many real-world scenarios, labeled training data were in short supply or can only be obtained with expensive cost. This problem has become a major bottleneck of making machine learning methods more applicable in practice (Yang et al., 2020).

In the last decade, semi-supervised learning (Blum & Mitchell, 1998) techniques were proposed to address the labeled data sparsity problem by making use of a large amount of unlabeled data to discover an intrinsic data structure to effectively propagate label information.

Nevertheless, most semi-supervised methods require that the training data, including labeled and unlabeled data, and the test data were both from the same domain of interest, which implicitly assumes the training and test data were still represented in the same feature space and drawn from the same data distribution. Instead of exploring unlabeled data to train a precise model, active learning, which is another branch in machine learning for reducing annotation effort of supervised learning, tries to design an active learner to pose queries, usually in the form of unlabeled data instances to be labeled by an oracle (e.g., a human annotator). The key idea behind active learning is

that a machine learning algorithm can achieve greater accuracy with fewer training labels if it is allowed to choose the data from which it learns (Lewis).

However, most active learning methods assume that there is a budget for the active learner to pose queries in the domain of interest. In some real-world applications, the budget may be quite limited, which means that the labeled data queried by active learning may not be sufficient enough to learn an accurate classifier in the domain of interest.

Transfer learning, in contrast, allows the domains, tasks, and distributions used in training and testing to be different. The main idea behind transfer learning is to borrow labeled data or extract knowledge from some related domains to help a machine learning algorithm to achieve greater performance in the domain of interest (Pan & Yang, 2009). Thus, transfer learning can be referred to as a different strategy for learning models with minimal human supervision, compared to semi-supervised and active learning.

There were four transfer learning approaches, including (Yang et al., 2020):

- Instance-transfer: Re-weighting the labeled data for the target domain.
- Feature-representation-transfer: Selecting a good feature set to reduce the difference between two domains.
- Parameter transfer: Discovering parameters in one domain and reusing these parameters in the target domain.
- Relational-knowledge-transfer: Mapping of knowledge between two domains.

With respect to the nature of the target task (Pan & Yang, 2009) distinguish the following three settings:

- Inductive transfer learning: target task is different than the source one.
- Transductive transfer learning: target task and source task were the same, but the domains were different.
- Unsupervised transfer learning: target task is different than the source task, but they were related to each other.

For deep neural networks, in some cases there may not be enough data to train the network or creating the labeled data might be expensive. transfer learning can be applied to adopt the knowledge that has been learned in earlier settings. For example,

there were various CNN models such as AlexNet (Krizhevsky et al., 2012), GoogleNet (Szegedy et al., 2015), and VGG (Simonyan & Zisserman, 2015), which can be used later on for similar tasks. The two common transfer learning strategies in deep learning were deep feature extraction and fine-tuning (Kaya et al., 2019).

In the deep feature extraction, the input data was provided to the pre-trained network and activation values of various layers were stored and used as features. In fine-tuning, deep neural network was trained for a similar task, in which labeling is relatively easier. While the first layers of the pre-trained network can be fixed, fine-tuning can be done on the final layers of the model to learn the properties of the new dataset. The pre-trained model was re-trained with the new small dataset and weight values of the model were updated according to a new task. Fine-tuning process occurs on the network using back propagation with labels. Learning to transmit is often faster than training a new neural network because all the parameters in the new network were not estimated from scratch. In the lower layers of the network, more general features exist such as color blobs and Gabor filters and they can be transferred to other tasks as well. However, in higher layers, features were more task specific. Deep learning systems provide high performance for several problems, but they require huge amount of data and time for their training. In this case, reusing these pre-trained models for similar tasks is quite helpful.

Deep convolutional neural networks have a wide range of applications in image feature extraction (Han et al., 2017). However, with the continuous improvement of the network level, the feature dimension of image extraction is also rising, which makes the subsequent processing of features extracted by deep convolutional neural networks rely heavily on the dimension reduction algorithm.

There were a few CNN models that were successfully trained for this face recognition task such as VGGFace. Its architecture proposed by Parkhi et al. (2015) which achieved state-of-the-art results in extracting features and relied on a very deep facial recognition CNN architecture. It consists of 5 convolution blocks and 3 fully connected layers including fc6, fc7, and fc8 as shown in Fig. 4-3. Each convolution block comprises of two or three convolutional layers with a max-pooling layer to reduce the size of the output feature map.

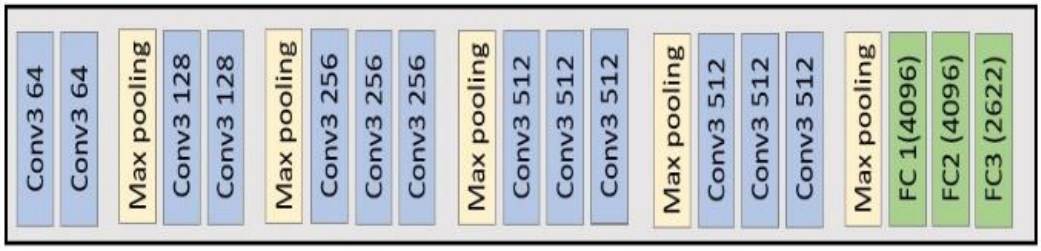

Fig. 4-3. The original VGGFace pre-trainer adopted for deep face recognition (Parkhi et al., 2015).

Achieving an optimal pain detection algorithm is a challenging task. There is a great difference between the target task's image set and the pre-trained image set, regardless of the number of categories or image styles. In the retrieval task of the target image set, the visual features of the image were directly extracted by the pre-trained CNN model. to make the pre-trained CNN model parameters more suitable for the feature extraction of the target image set, the VGGFace pre-trained CNN model was fine-tuned as follows: VGGFace were used and retrain it for pain estimation by keeping the convolution layers of this model unchanged while replacing the fully connected layers with a new fully connected layer. 5 convolutional blocks and the new replaced fully connected layer were retrained by transfer learning. The replaced, fully connected layer was followed by dropout and the size of it was 1024. Fig. 4-4 shows the proposed fine-tuned VGGFace architecture to extract features from image pain data.

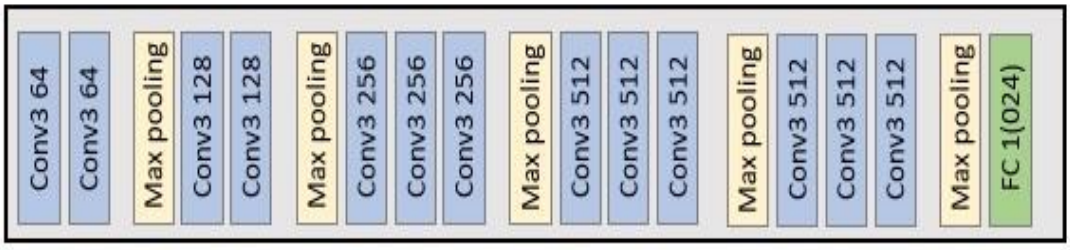

Fig. 4-4. The proposed fine-tuned VGGFace architecture to extract image feature.

The ADAM-optimizer is one of the most popular gradient descent optimization algorithms and faster than other optimizers. The ADAM-optimizer were applied to retrain fine-tuned VGGFace since it is a superior optimizer that can tune the parameter automatically during training (Han et al., 2018).

in this research paper, there were a total of 38,816 features, which were extracted from the training data set, calculated according to the input shape of the extracted features.

For the training data set, these were denoted as (9704, 4) where the number 9704 refers to the number of training images and so, we were able to obtain a product $9704 \times 4 = 38{,}816$. the 4 distinct output features (per image) extracted from the fine-tuned VGGFace were transferred into the PCA algorithm with an aim to reduce the dimensionality of the extracted features and also to speed up the classification algorithm. the proposed new feature extraction model applied pretrained and finetuned VGGFace and then the PCA algorithm was used to achieve dimension reduction. In the following sections the components of the proposed feature extraction model were explained. Fig. 4-5 shows the proposed deep feature extraction framework contains finetune VGGFace and PCA.

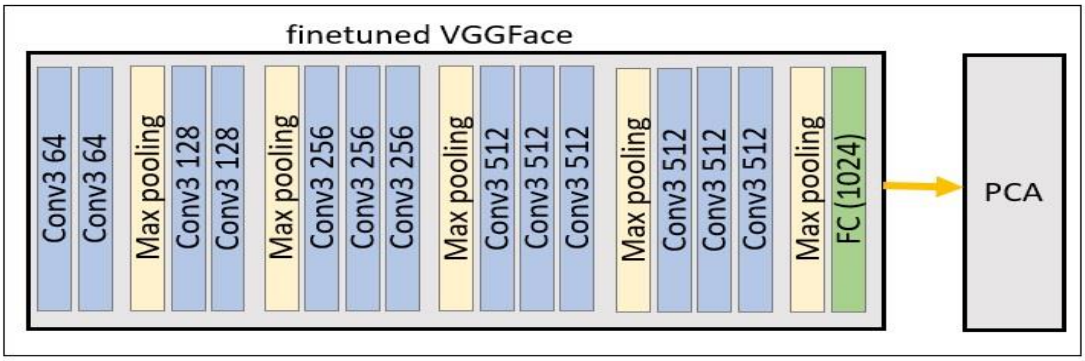

Fig. 4-5. The proposed deep CNN-PCA feature extraction framework.

The PCA algorithm with an aim to reduce the dimensionality of the extracted features and to speed up the classification algorithm was used. PCA is a dimensionality reduction method that is useful in different applications such as image compression, facial feature extraction, face recognition and finding patterns from large dimensional images (Damale & Pathak, 2018; Li et al., 2009). It helps to choose the best set of data dimensions that will make the model perform better, and to speed up the algorithm performance (Sun et al., 2014). PCA (Ueda & Hoshiai, 1997; Xu et al., 2019; Zhang et al., 2019) is a general-purpose dimension reduction and data analysis tool, which is mainly used in important research fields such as pattern recognition, artificial intelligence and data mining. The essence of PCA is to project data samples in high-dimensional space into low-dimensional space by linear transformation while preserving the original data features as much as possible (Ma & Yuan, 2019). For example, when judging the category of an image, color histograms, texture features, edge shapes, or region shapes may be considered. If all the features were retained, there may be hundreds, so it is necessary to preserve the main features of the image

and then replace the original features with a linear combination. The PCA algorithm mainly achieves the dimension reduction of the original image by retaining components with large variance and large amount of information and removing components with small variance and insufficient information.

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. For doing this, the explained variance for each component was calculated by Python software. The dimensionality reduction process was achieved through an orthogonal, linear projection operation. The applied PCA operation can be defined as (Goodfellow et al., 2016):

$$Y = XC \tag{4-2}$$

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 was to find the projection matrix

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

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

$$X = \cup \sum \vee^{T} \tag{4-3}$$

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,

λ_n, for n = 0,⋯ ,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

$$V = [v_1, \dots, v_N] \tag{4-4}$$

and

$$C = [c_1, \dots, c_P] \tag{4-5}$$

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 were the standard deviations of *X* along the principal directions in the space spanned by the columns of *C*. 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 (Goodfellow et al., 2016):

$$R_{cev} = \frac{\sum_{n=1}^{P} \lambda_n^2}{\sum_{n=1}^{N} \lambda_n^2} \tag{4-6}$$

Fig. 4-6 describes that selecting 2 components was able to 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 were 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 78% of the variance, while we need around 2 components to describe close to 100% of the variance.

if

$\rho =$ vectors of length of number of frames ,

and

$f =$ number of features after PCA reduction

then

$$\rho \times f \tag{4-7}$$

used as potential inputs were passed into the EJH-CNN-BiLSTM algorithm. In this case, based on the above discussion,

$f = 2$

and so, length of the number of frames for the ConvD1 layer

$(\rho = 2)$

with the input shape (2, 2) for the ConvD1 layer were selected. It was also found that

ρ = 20

worked relatively well for BiLSTM algorithm and the input shape for the BiLSTM algorithm was selected according to (20 ,2) in order to enter the EJH-CNN-BiLSTM classifier system.

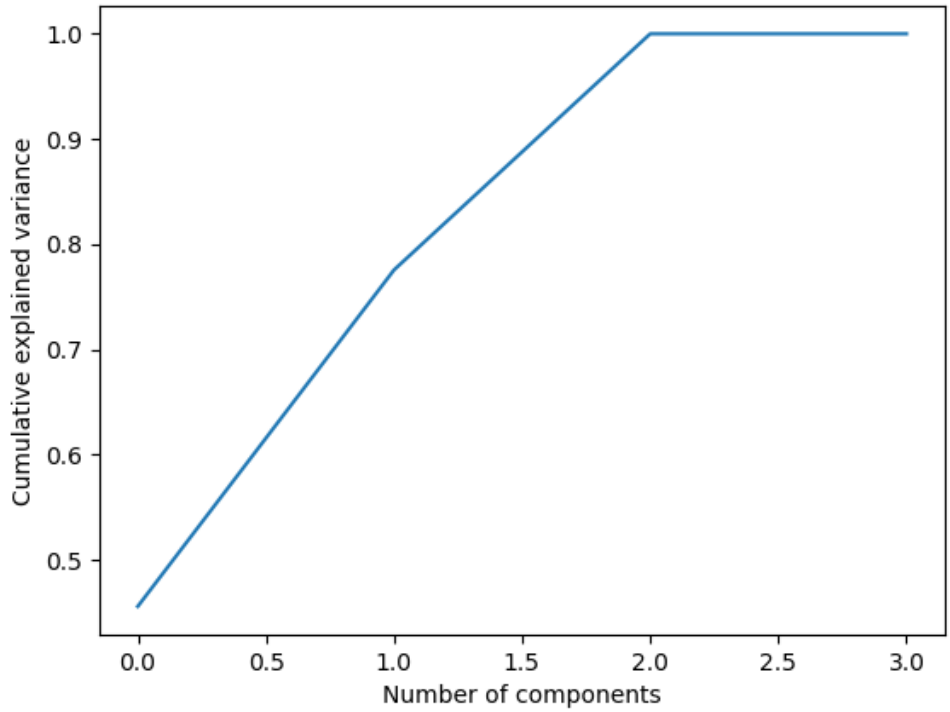

Fig. 4-6. PCA explained variance ratio for four components

4.3 EJH-CNN-BiLSTM Classifier

A new hybrid deep learning classifier denoted as the EJH-CNN-BiLSTM was developed to classify pain levels in multi classifications. The extracted and reduced features from the VGGFace-PCA were transformed into the new developed EJH-CNN-BiLSTM classifier. The EJH-CNN-BiLSTM consists of two streams hybrid deep learning model which their outputs joint as indicated in the classification section of the Fig.7. The CNN-BiLSTM hybrid deep learning contains two streams. At first, the selected features were transferred into CNNs includes two one-dimensional convolutional neural network (Conv1D) (Gulli & Pal, 2017) layers which outputs of Conv1D-1 were transferred into Conv1D-2 as inputs. Then, the outputs of the ConvD1-2 were entered to both the BiLSTM1 (stream 1) and BiLSTM2 (stream 2) separately. Finally, the outputs of both the BiLSTM1 and BiLSTM2 were merged and the Gaussian noise calculated by PCA applied to classify four pain levels.

Structure of applied CNNs: Convolutional Neural Networks were extremely powerful models often used in the space of computer vision. CNNs were fast and Conv1Ds have also shown success on sequential learning problems and continue to be explored in this new space. CNN is a kind of deep learning technique, which has become very popular in the field of image understanding. The basic convolutional neural network is mainly composed of input layer, convolutional layer, pooling layer, fully connected layer and output layer. The input to a CNN is an $n \times n \times m$ image, where n is the height and width and m is the number of channels, and there will be k convolutional filters of size $a \times a$ in the convolutional layer, where $a < n$.

Fig. 4-7 shows the convolutional layer and the pooling layer incorporated to form a plurality of convolutional groups, and the features were extracted layer by layer, and finally the classification was completed through several fully connected layers. Convolutional layer is the core part of the network, which is mainly used to extract the local detail information of the image. The pooling layer is used to down-sample the computed feature map from the convolutional layer, which can reduce features while preserving the local invariance of features. Pooling operations generally include spatial pyramid pooling, average pooling, random pooling, and max pooling. The fully connected layer is used to encode the three-dimensional feature map into a one-dimensional vector. Finally, a multi classifier was connected to the network, and the

corresponding error gradient was computed before backpropagation was used to update the neural network.

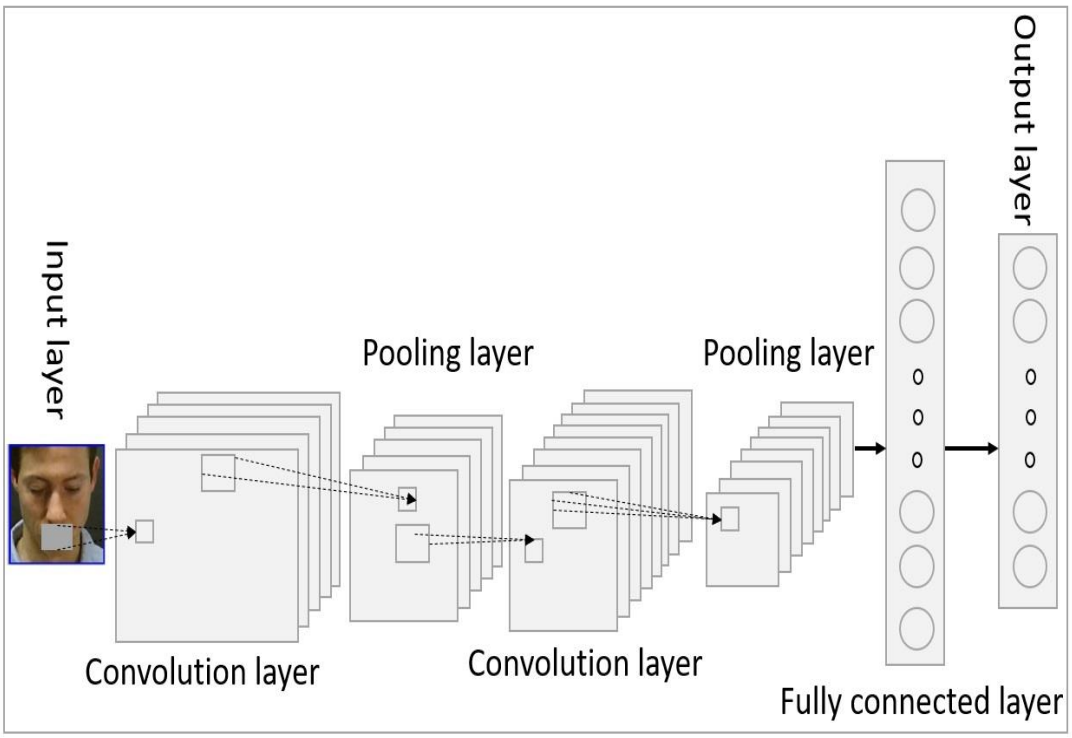

Fig 4-7. General architecture of the convolutional neural network (CNN).

The characteristics of convolution neural network, such as local connection, weight sharing and pooling operation, can effectively reduce the complexity of the network and reduce the number of training parameters, so that the model has a certain degree of invariance to the translation, distortion and scaling. It is robust and easy to train and optimize. Based on these superior features, it is widely used in various signal and image processing tasks (Liu et al., 2019).

Table 4-1 indicates the structure of applied Conv1D-1 and Conv1D-2 in EJH-CNN-BiLSTM.

Table 4-1. The structure of Conv1D-1 and Conv1D-2 used in the EJH-CNN-BiLSTM proposed model.

Layer name	Input shape	Filter size	Kernel size	Layer parameters	Padding
Conv1D-1	length = 2 feature = 2 input shape = (2,2)	256	10	Activation: ReLU	Same
Conv1D-2	length = 2 feature = 2 input shape = (2,2)	128	10	Activation: ReLU	Same

Structure of applied RNNs: Since the data type is video and contains video image frames, the RNN is used in this model to improve classification. RNNs were suited to sequential data since their neurons have connections (weights) between the next layers and keep information from previous inputs. BiLSTM as an RNN type has an elegant solution for each sequence forward and backward as two separate hidden states to capture past and future information, respectively.

LSTM deep learning was 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 (Gers & Schmidhuber, 2001; Schmidhuber, 2015). Fig. 4-8 shows the architecture of an LSTM cell, in which the cell state part was calculated by:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (4\text{-}8)$$

The output of the forget gate was calculated as:

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (4\text{-}9)$$

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

$$c_t = f_t c_{t-1} + i_t tanh(W_{xc}x_t + w_{hc}h_{t-1} + b_c) \quad (4\text{-}10)$$

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:

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_o c_t + b_o) \quad (4\text{-}11)$$

$$h_t = o_t tanh(c_t) \quad (4\text{-}12)$$

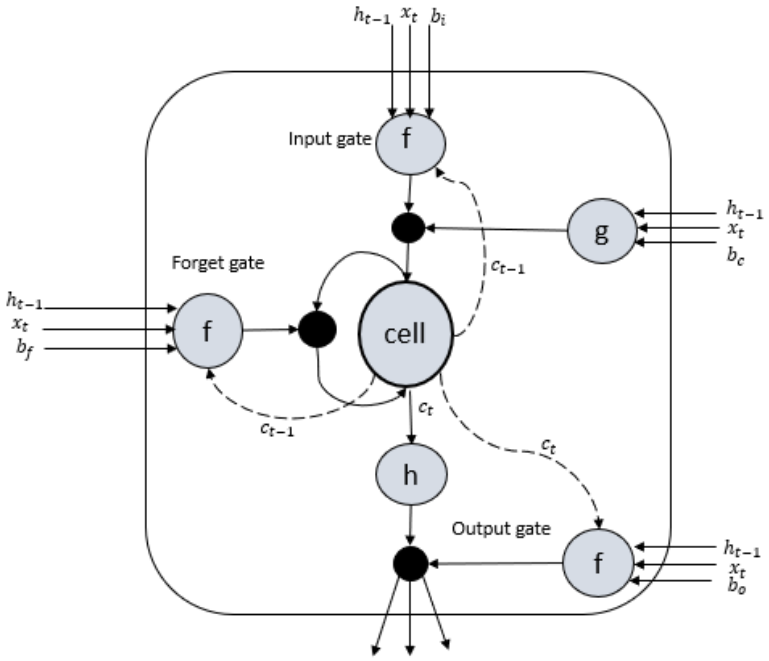

Fig. 4-8. The architecture of an LSTM unit (Gers & Schmidhuber, 2001; Schmidhuber, 2015). Inputs: x_t: Input vector, c_{t-1}: memory from previous block, h_{t-1}: output of previous block, b: Bias Outputs: h_t: the output of current block, c_t: memory from the current block

Obtaining both past (left) and future (Selltiz et al., 1976) frames is useful and essential for sequences labelling tasks such video analysis for pain recognition from facial expression images. However, the LSTM's hidden state h_t takes information only from the past frame, without having information from the future frame. BiLSTM (Dyer et al., 2015) as an elegant solution presents each sequence forwards and backwards as two separate hidden states to capture past and future information, respectively.

The experimental results show incredible improvement in compared with model which used only two streams BiLSTM for this classification problem. Table 4-2 shows the structure of the BiLSTM1 and BiLSTM2.

Table 4-2. Structures of the BiLSTM1 and BiLSTM2 used in the EJH-CNN-BiLSTM proposed model.

Layer name	Layer parameters
BiLSTM1	Input shape = (20,2) Filter = 256 Dense = 4096 Activation = ReLU Gaussian Noise = PCA-Std Dropout = 0.5
BiLSTM2	Input shape = (20,2) Filter = 32 Dense = 4096 Activation = ReLU Gaussian Noise = PCA-Std Dropout = 0.5

Merging and applying the Gaussian noise: The gaussian noise calculated from PCA added to outputs of the dense layers during the training of the proposed EJH-CNN-BiLSTM framework. Several studies report the addition of noise during neural networks training as a tool to improve the generalization capability and convergence time. The previous studies focused on creating new input patterns by adding random noise drawn from a Gaussian distribution increased generalization power if the amount of noise is kept sufficiently small to have no disruptive effect on the desired output. gaussian noise was added to the PCA components and calculated by *Numpy.STD* of *Keras* library in *Python* and the calculated amount added to outputs of the dense layers**.** The *Numpy.STD* computes the Standard Deviation (SD) of the given data. SD was measured as the spread of data distribution in the given data set.

$$SD = \sqrt{mean(abs(x - x.mean())^2)} \quad (4\text{-}13)$$

Dense Layer and compiling the model:

Some algorithms such as Logistic Regression, Perceptron, Support Vector Machines were designed for binary classification problems. They cannot be used directly for multi-class classification task. some heuristic methods such as One-vs-Rest (OvR) and One-vs-One (OvO) for these algorithm can be used to split a multi-class classification problem into multiple binary classification datasets and train a binary classification model each (Wang et al., 2010). However, in deep learning the softmax activation

function which was used in the output layer to handle multi-class problems (Goodfellow et al., 2016).

The *softmax* function takes as input a vector *z* of *K* real numbers and normalizes it into a probability distribution consisting of *K* probabilities proportional to the exponentials of the input numbers. Some vector components could be negative, or greater than one; and might not sum to 1 prior to applying *softmax*, but after applying *softmax*, each component will be in the interval (0,1). The *softmax* function was defined by the following formula (Goodfellow et al., 2016):

$$\sigma: R^K \rightarrow R^K \tag{4-14}$$

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \; for\; i = 1, \ldots, K \; and \; z = (z_1, \ldots, z_k) \in R^K \tag{4-15}$$

Output set to calculate for *four classes* with activation *Softmax*. The network was optimized with Adam optimizer since it has proved to be more stable than Stochastic Gradient Descent (SGD) (Han et al., 2018). In training *loss* selected as *categorical_crossentropy* to calculate accuracy, mean squared error (MSE), and mean absolute error (MAE) of the proposed algorithm in measuring pain level in 4 classes.

4.4 Proposed EJH-CNN-BiLSTM Algorithm

The details of the proposed EJH-CNN-BiLSTM model summarized in Algorithm 4-3. Five epochs and 48 batches to train and test the proposed algorithm used.

Algorithm 4-3: EJH-CNN-BiLSTM algorithm

```
1   procedure EDLM (input, n, j, batch)
2       Pre-process (input)
3       for k ← 0, n do
4           finetune (VGGFace)
5           for epoch ← 0, j do
6               features ← train (finetune (VGGFace))
7           end for
8           SF ← PCA (features)
9           GN ← Calculate (GN)
10          for epoch ← 0, j do
11              o1 ← CNN-BiLSTM1(SF)
12              o2 ← CNN-BiLSTM2(SF)
13              out ← merge (o1, o2,)
14              out ← GN (Yang et al.)
15              train-test (model (SF, out))
16          end for
17      end for
18  End procedure
```

4.5 Experimental Results

The pre-processed selected dataset from the UNBC-McMaster Shoulder Pain database was divided based on k-fold cross validation technique which k=10. For training and testing the fine-tuned VGGFace, five epochs and 48 batches was used. During learning the fine-tuned VGGFace the accuracy level increased for each batch size gradually and the loss amount decreased. Fig. 4-9 shows the increasing accuracy and decreasing loss amount during learning time for feature extracting from finetuned VGGFace for UNBC-McMaster Shoulder pain dataset. In epoch 5 the accuracy level reach to the best by approximately 90%. The blue line in the Fig. 15 indicates the increasing accuracy level during the learning time of the finetuned VGGFace. In contract, the red line in the same figure indicated the decreasing of the loss by increasing epoch.

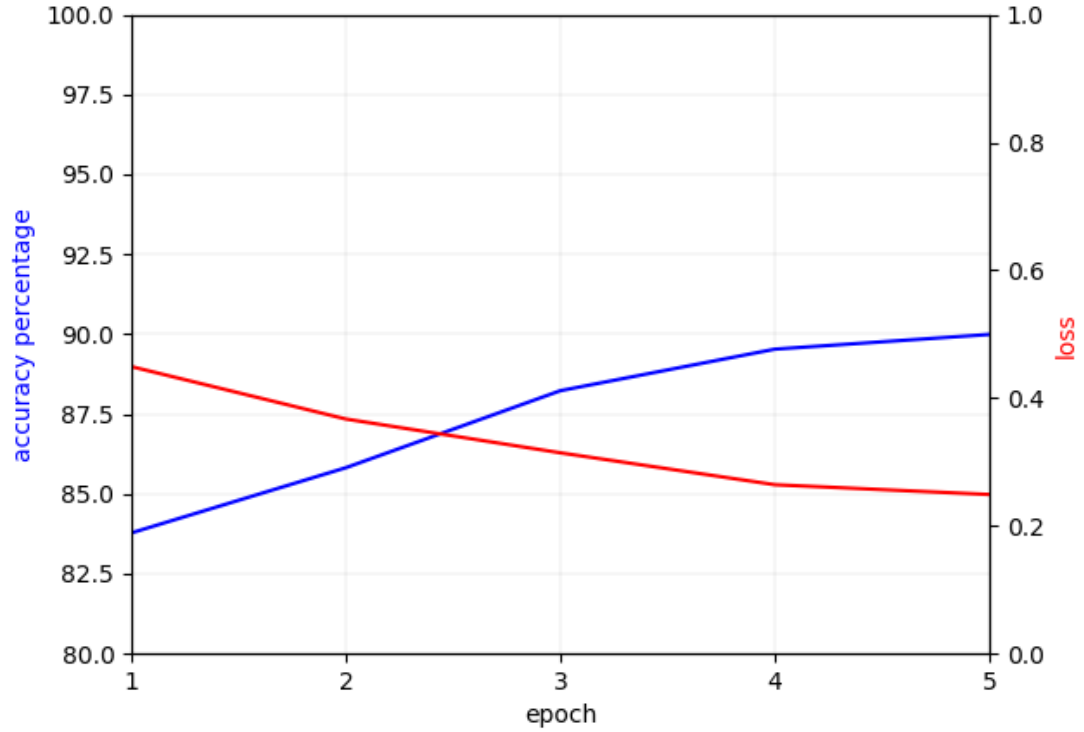

Fig. 4-9. The accuracy and loss level during finetuned VGGFace feature extracting learning for UNBC-McMaster Shoulder Pain dataset.

The obtained results from the finetuned VGGFace were compared with different pre-trainer such as VGG16, GoogleNet, Alexnet. Table 10 compares the accuracy level of the different CNN pre-trainer accuracy applied in the UNBC-McMaster database. As it is explained in Table 4-3, the fine-tuned VGGFace has better performance.

Table 4-3. The accuracy results for various pre-trainers applied for feature extraction task for the UNBC-McMaster Shoulder Pain dataset.

Pre-trainer	**Accuracy for fine-tuned model (%)**
VGG16	83
GoogleNet	77
Fine-tuned VGGFace	88
AlexNet	75

Table 4-4 shows the average of training MSE, training MAE, training accuracy, test MSE, test MAE, test accuracy, and AUC of the average for 10-fold cross-validation performance measurement. The average accuracy of 91.2% for the training set, 90% accuracy for the testing set, and 98.4% for AUC were achieved.

Table 4-4. The average performance of the EJH-CNN-BiLSTM on the UNBC-McMaster Shoulder Pain database for 10-fold cross validation.

Training MSE	**Training MAE**	**Training Accuracy (%)**	**Test MSE**	**Test MAE**	**Test Accuracy (%)**	**AUC (%)**	**PR curves (%)**
0.03	0.06	91.2	0.04	0.07	90	98.4	98

the proposed EJH-CNN-BiLSTM evaluated for each class based on TP, F-measure, precision, and AUC in 10-fold cross-validation. Table 4-5 shows the average performance measuring of the proposed algorithm for each pain levels.

Table 4-5. The average performance of the EJH-CNN-BiLSTM on the UNBC-McMaster Shoulder Pain database per each class.

Class	**TP (%)**	**f-measure (%)**	**Precision (%)**
No pain	87.70	87.51	87.5
Weak pain	88.50	89.10	90.02
Mild pain	90.30	87.93	86
Strong pain	93.20	95	96.54

The obtained results were compared with the baseline models (See Table 4-6). The results indicate that the proposed model has significant performance improvement since its AUC and accuracy was higher than other models mentioned in the Table 16. All of them were tested in the same selected balanced dataset with total 10783 images by applying the 10-fold cross-validation technique for 25 subjects.

Table 4-6. Comparing the performance of the proposed model with different versions of the deep learning algorithm designed during experimental test based on average amount of accuracy and AUC for 10-fold cross validation on the UNBC-McMaster Shoulder Pain database.

No	Model	Accuracy (%)	AUC (%)	PR curves (%)
1	CNN	54.45	44.8	46
2	VGGFace+CNN	57.3	52	58
3	VGGFace+BiLSTM1	65	75.3	77
4	VGGFace+BiLSTM1+BiLSTM2	73	78.5	82
5	*EJH-CNN-BiLSTM*	*91.2*	*98.4*	*98*

The 10-folds cross-validation was not a subject-independent performance measurement since images from the same subject may appear in both the training and testing sets. To control for this limitation, LOOCV technique was applied in which subjects of the training were removed from the testing. This validation allows exploring how the proposed method for pain intensity measurement generalizes to a new set of subjects who were not part of the training set. The LOOCV consists of building 25 classifiers for each one of the four levels of pain intensity and iterating the process. Table 4-7 shows the achieved results and compared them with state-of-the-art results for the same task by using LOOCV.

As it is shown in the Table 4-7, Lucey, Cohn, Prkachin, et al. (2011) and Lucey, Cohn, Matthews, et al. (2011) applied all images of the UNBC-McMaster Shoulder Pain database for binary classification however, the remined research works were shown in the table, have applied a balanced dataset of the same database. As discussed in Section 3.3.4.1 and illustrated in Figure 3.4, the original database labels were unbalanced and it is very challenging to perform the modelling experiments especially for multi-classes classification problem. later the other research works and, in this thesis, balance database by oversampling and under sampling techniques which is called in the Table 4-7 as *Down-up were* used.

The MIntPAIN database includes pain video data taken by electrical stimulation in five levels (Level 0 – no pain to Level 4 – highest pain level) of 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 the 20 subjects was selected. The number of no pain video sequences were more than any other. based on the specific character of the database it is likely that any model would 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 is balanced using under resampling techniques to reduce the majority class (no-pain class); some no pain sequences were removed. The resampling technique was applied on the selected dataset since a few subjects were missing for some sweeps and there is also not an equal proportion for each class. the under-sampling technique was applied to reduce the majority class, and some no painful sequences (label 0) were removed. 34800 video frames were selected for experimentation in this research.

Table 4-7. Comparison of the proposed EJH-CNN-BiLSTM model's results with the state-of-the-art results in the UNBC McMaster Shoulder Pain database to detect pain from facial expressions based on LOOCV.

Ref	**Level**	**Classifier**	**AUC (%)**	**Accuracy (%)**	**F-measure (%)**	**MSE (%)**	**MAE (%)**	**Size of data**
(Lucey, Cohn, Prkachin, et al., 2011)	2	SVM	83.9	-	-	-	-	All
(Lucey, Cohn, Matthews, et al., 2011)	2	SVM	84.7	-	-	-	-	All
(Rodriguez et al., 2017)	2	CNN-LSTM	93.3	83.1	-	74	50	Down-up
(Bellantonio et al., 2016)	3	CNN-RNN	-	61.9	-	-	-	Down-up
(Hammal & Cohn, 2012)	4	SVM	-	80	60	-	-	16657 image
The proposed model	*4*	*EJH-CNN-BiLSTM*	*88.7*	*85*	*78.2*	*20.7*	*17.6*	*10783 image*

The comparison results of the proposed EJH-CNN-BiLSTM model with the-state-of-the art results show the proposed algorithm is significantly more effective. In terms of error measuring, it has fewer errors measured by MAE and MSE in comparison with results obtained by (Rodriguez et al., 2017). Furthermore, it achieved the highest accuracy by 85% in comparison with the results of the other hybrid deep learning algorithms done by (Bellantonio et al., 2016; Rodriguez et al., 2017; Zhou et al., 2016). However, the AUC achieved by (Rodriguez et al., 2017) is higher than our proposed model. the comparison results show the EJH-CNN-BiLSTM obtained high performance in accuracy and f-measure in comparison with the results achieved by (Hammal & Cohn, 2012) tested in four pain levels. The other noticeable point of the results of the proposed model is its ability in the four-levels classification.

The proposed EJH-CNN-BiLSTM algorithm is efficient in terms of running time. The PCA used in feature selection of the algorithm accelerates the algorithm during training and testing. The running time of the algorithm improved speed by up to 3 hours for the whole process by in Core i7 computer with 16 GB RAM. Fig. 4-10 shows the algorithm running time before and after using the PCA.

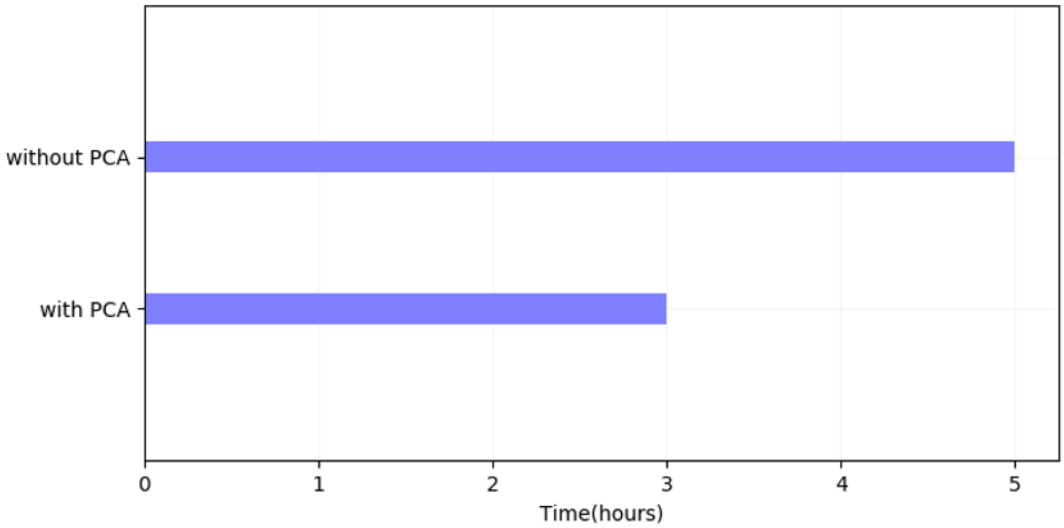

Fig.4-10. Comparing the running time of the EJH-CNN-BiLSTM with or without PCA.

4.6 Discussion

The analysis of the obtained results indicates that the involvement of fine-tuned pre-training and the proposed EJH-CNN-BiLSTM method, when combined with the PCA with additive Gaussian noise can improve the accuracy of the algorithm. By comparing and analyzing the obtained results, we can conclude as follows: In terms of small datasets, CNNs get very low classification accuracy. There is a large number of parameters that have not been fully trained. pre-training and fine-tuning were very effective transfer learning techniques for image classification. As the results show, the fine-tuning network can increase the accuracy of the algorithm. In terms of efficiency, PCA reduces the dimensionality of the selected features then accelerates the algorithm's running time. Using PCA for dimensionality reduction involves zeroing out one or more of the smallest principal components, resulting in a lower-dimensional projection of the data that preserve the maximal data variance. The PCA with additive Gaussian noise significantly improves the performance of the algorithm.

4.7 Chapter Summary

A novel hybrid joint CNN-BiLSTM deep learning approach for four level pain recognition on facial images is proposed. To achieve satisfactory results in terms of pain intensity estimation, the fully connected layer of the VGGFace was improved for this task by adding an extra fully connected layer and the dimensionality of the extracted features reduced by PCA to increase the overall computational efficiency of the proposed algorithm. The reduced extracted features, which were the most useful patterns for pain intensity estimation, feed to the classification section of the newly developed EJH-CNN-BiLSTM model. experimental results demonstrated that the proposed EJH-CNN-BiLSTM method significantly improves the performance achieved by using the conventional approach. The enhanced algorithm obtained an AUC of 98.4% and test accuracy of 90% on the balanced UNBC-McMaster Shoulder Pain database. Furthermore, for generalizing the proposed algorithm and comparing it with other similar research works, the leave-one-subject-out performance measuring technique was applied as well, and obtained results indicate the effectiveness of the proposed algorithm for unseen data. The artificial intelligence method developed in this study can have useful implications for the medical diagnostic areas, particularly,

supporting the implementation of automatic pain management practices for clinicians and other medical researchers.

CHAPTER 5

The Proposed Ensemble Deep Learning Model (EDLM)

The proposed ensemble deep learning model, developed in this chapter, consists of two newly developed sections including image-preprocessing, feature extracting and EDLM classifier. A stock ensemble CNN-RNN network in three streams was developed which their outputs were merged. The proposed approach was trained and tested in two databases including the MIntPAIN (Haque et al., 2018) labeled in VAS and the UNBC-McMaster Shoulder Pain Archive Dataset (Lucey, Cohn, Prkachin, et al., 2011) which labelled the video frames in PSPI and FACS metrics. The novelty and contributions of this research work were as follows:

1. A new approach including a three-streams ensemble CNN-RNN classifier which their outputs were merged as the late fusion was assembled to classify pain in five levels from extracted features.
2. The whole proposed framework as known Ensemble Deep Learning Model (EDLM) model in this study was trained and tested in two popular painful face databases and the obtained results indicates the proposed model has high performance in comparison with the state-of-the-art techniques and baseline models.

In the following, a brief description of the ensemble learning explained and then the proposed EDLM classifier is elaborated.

5.1 Ensemble Deep Learning

Ensemble Learning Systems (ELSs) were inspired from an innate behavior of humans; the opinions of several experts were being collected for making a decision and then, based on these opinions, the final decision is made, especially if these decisions lead to financial, social, and medical consequences (Mousavi & Eftekhari, 2015). This learning method is useful in several cases, including online learning, incremental learning, fusion data, feature selection, and confidence estimation. Ensemble learning is an effective method and can improve the generalization ability of classification. 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 (Freund & Schapire, 1997). Ensembles of learning algorithms were effectively used in many computer vision problems to improve the classification performance (Minetto et al., 2019; Simonyan & Zisserman, 2015).

According to Dietterich (2000) 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". Ensemble learning algorithms improve the generalization ability. Ding and Tao (2017), used ensemble CNN for video-based face recognition. Their model outperforms previous approaches such as Deep Face (Taigman et al., 2014), DeepID2+ (Sun et al., 2015), and VGGFace (Parkhi et al., 2015). According to SHARKEY (1996), 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 (Hansen & Salamon, 1990). Xie et al. (2013), proposed Horizontal and Vertical Ensemble methods 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.

Ensemble learning is composed of three different main parts: sample selection, training the base classifiers to compose the Base Classifier Pool (BCP), and combining the BCP; as ensemble learning decreases the risk of selecting a single classifier with a weak performance, it improves the classification accuracy in comparison with single classifier. Numerous ensemble-based algorithms were proposed, that the most common of them were bagging, boosting, stacking, and random forest.

Stacking Wolpert (1992) is a learning approach based on ensemble learning which combines the predictions made by multiple base classifiers generated by using different learning algorithms $L_1, L_2, \dots, L_n$. These classifiers were trained on the same training data D_{train} containing examples in the form $s_i =< x_i, y_i >$, where x_i is the input vector, and y_i is the class label assosiated with it. In the first phase, base classifiers $L_1, L_2, \dots, L_n$ make predictions for the query instance x_q. In the second

phase, the meta-classifier *M* combines the predictions made by base classifiers and predicts the final class label.

5.2 Proposed EDLM Classifier Structure

The novelty of this study is to propose a new ensemble deep learning model (EDLM) to classify pain intensity in multi-levels from facial expression video frames data. The proposed framework has three steps including, image pre-processing, feature extracting, and EDLM classifier. The block diagram of the proposed system is shown in Fig. 5-1.

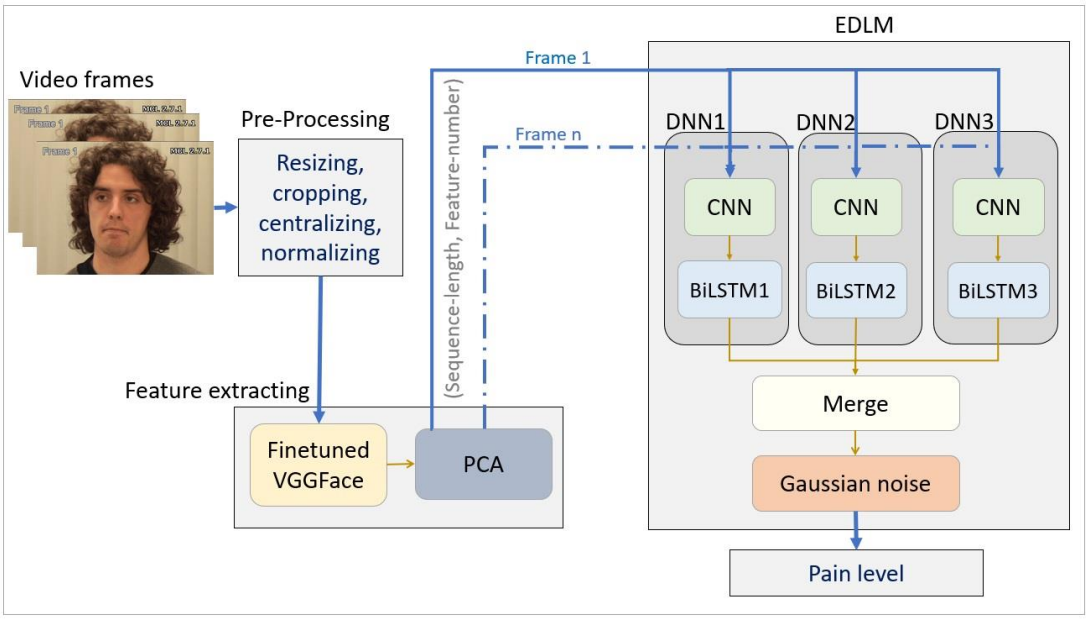

Fig. 5-1. Block diagram of the proposed ensemble deep learning model (EDLM) to detect pain in multi-classes from facial expressions.

The selected dataset is 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. the OpenCV face recognition algorithm was used to detect faces from noisy pictures. Then, face detected images were cropped and centralized. 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 as describes in

Section 4.1, Algorithm 4-1, and Algorithm 4.2. Fig. 5-2 shows the pre-processed video frames of the MIntPAIN database.

Fig. 5-2. Examples of video frames per 5 levels after removing backgrounds, cropping, and resizing.

For MIntPAIN database, 125280 features were extracted from the training data set, calculated according to the input shape of the extracted features. For the training data set, these were denoted as (31320, 4) where the number 34800 refers to the number of training images and so, we were able to obtain a product $31320 \times 4 = 125280$. the 4 distinct output features (per image) extracted from the fine-tuned VGGFace were transferred into the PCA algorithm with an aim to reduce the dimensionality of the extracted features and to speed up the classification algorithm.

Fig. 5-3 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 were 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.

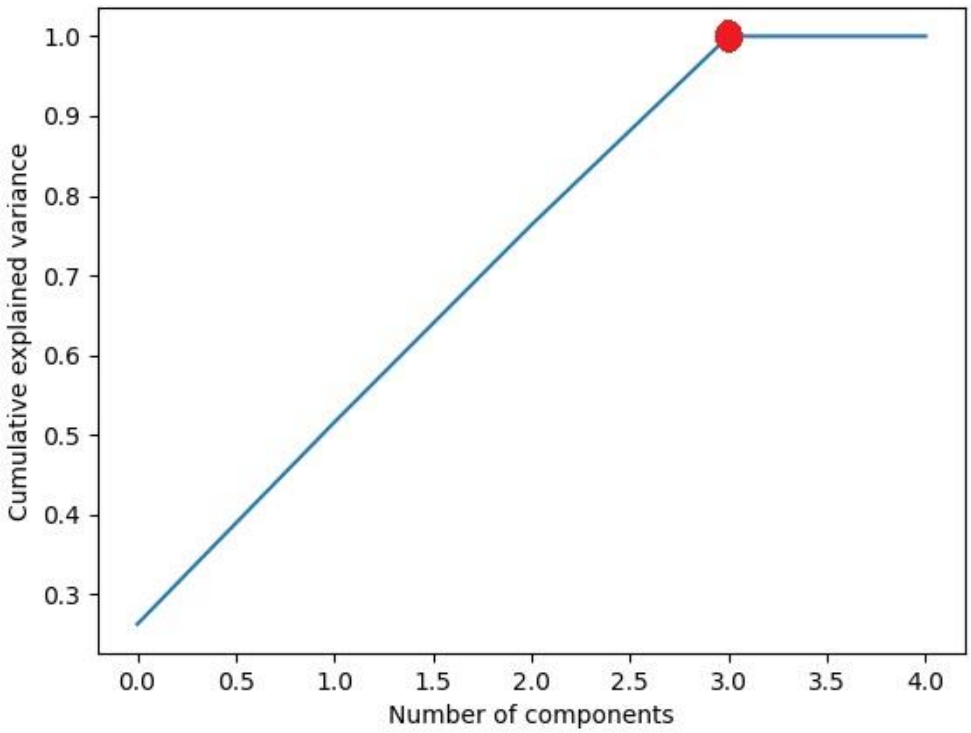

Fig.5-3. Number of components to select from extracted features by PCA for MIntPAIN database.

Then the pre-processed images enter the finetuned VGGFace-PCA feature extractor. The extracted and reduced features were transferred into the EDLM classifier for classification. The EDLM consists of three stream CNN+RNN deep learning network which were combined as stocked ensemble learning with merging their outputs. The proposed EDLM classifier as an ensemble deep learning network was developed in varying initial weights and network architecture. 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. The experimental results indicated that using three hybrid CNN+RNN has more accurate results than networks that only include RNN. These three independent and hybrid deep learning networks were DNN1, DNN2, and DNN3 which were developed using different parameter, weight, and architecture. The

configurations of these networks were described in Table 5-1. As can be seen from Table 15, DNN1 and DNN2 contain two CNNs with Conv2D architecture which their output shift in stack way to a BiLSTM. However, DNN1 and DNN2 were different in weighting. For DNN3, a different architecture of CNN+RNN was used. a CNN with Conv1D was selected and its output was transferred into a LSTM.

Table 5-1. 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

The simple average method (Akcay et al., 2018) was employed as merely a combining model for classification of pain levels from facial expressions. The output of SAM is the mean value of each member model's output. As the simplest combining model, the advantage of SAM is its ability to be at least better than the worst member model (Jeong & Kim, 2009).

$$y_c = \frac{1}{n_c}\sum_{k=1}^{n_c} y_k \tag{5-1}$$

$$\sigma_c^2 = var(y_c - y_0) = \frac{1}{n_c} = \sigma_{k,avg}^2 + \frac{n_c - 1}{n_c}\sigma_{k,j,avg} \tag{5-2}$$

$$\sigma_{k,avg}^2 = \frac{1}{n_c}\sum_{k=1}^{n_c} var(y_k - y_0) \tag{5-3}$$

$$\sigma_{k,j,avg} = \frac{1}{n_c(n_c-1)} \sum_{k=1}^{n_c} \sum_{j=1,j\neq k}^{n_c} cov(y_k - y_0 , y_j - y_0) \quad (5\text{-}4)$$

where

y_c is the output of *SAM* model,

y_k or y_j is output of member model k or j;

y_0 is the observed value;

n_c is the number of member models;

σ_c^2 is variance of classification errors for SAM;

$\sigma_{k,avg}^2$ is the average of the error variances of member models;

and $\sigma_{k,j,avg}$ is the average of the covariance between each classification.

The details of the proposed EDLM method were summarized in Algorithm 5-1. During experimentation optimization for the feature extraction section, the model ran by 50 epoch and 48 batches. However, for learning the classifier, 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 5-1: The proposed EDLM algorithm

```
Procedure EDLM (input, n, j, batch)
    Pre-process (input)
    for k ← 0, n do
        finetune (VGGFace)
        for epoch ← 0, j do
            features ← train (finetune (VGGFace))
        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
            train (model (SF, out))
        end for
    end for
end procedure
```

5.3 Results and Discussions

To establish the robustness of the proposed EDLM model, two databases were used includes the MIntPAIN database and the UNBC-McMaster Shoulder Pain dataset. Next, the evaluated results compared with the baseline model and the state-of-the-art research. In the following the results of two databases were detailed.

5.3.1 MIntPAIN Database's Results

The features were extracted and reduced by the finetuned VGGFace-PCA. 50 epochs were applied in learning of the feature extraction model to reach its best performance. Fig. 5-4 illustrates the accuracy and the loss error encountered in the feature extraction part of 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.5-4 the accuracy level reached to its highest level by 81% in epoch = 50. It started from 32% in epoch 1 and gradually 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 reached the lowest level by 0.18 in epoch 50.

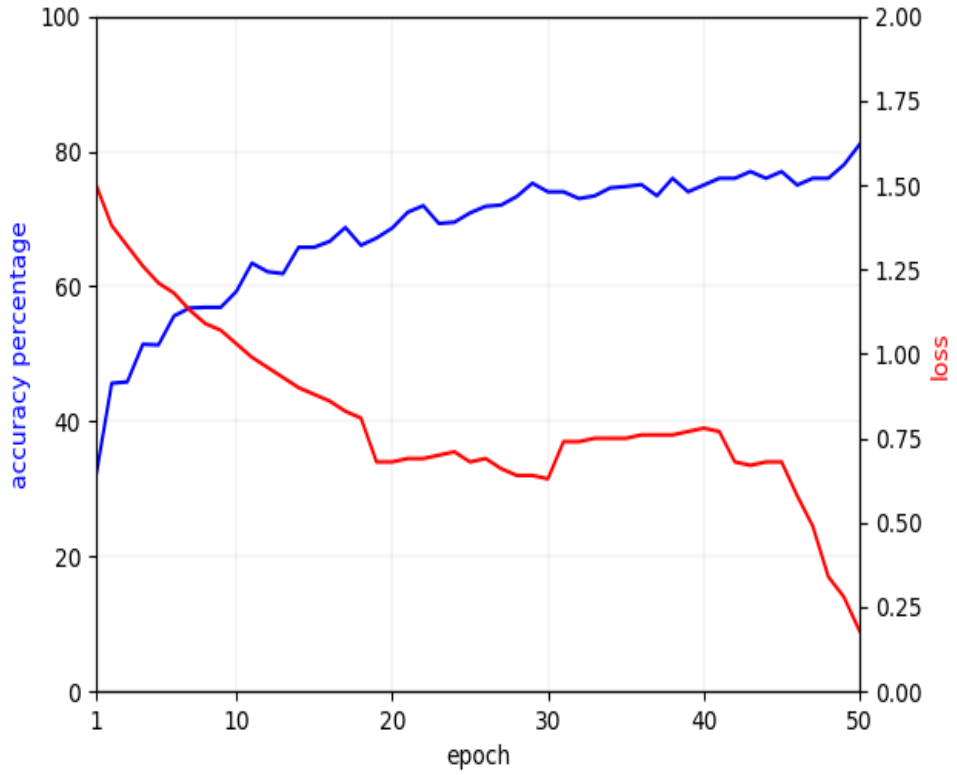

Fig. 5-4. Accuracy and loss error during 50 epochs in the early fusion of the EDLM model in the MIntPAIN database.

Later, the proposed classifier was trained and tested by selected features. Fig. 5-5 shows the accuracy and loss level during 5 epochs in average of 10 cross validation for late fusion. At first, accuracy started at 81% and then from the second epoch it reached 92.26% in epoch 5. The red graph in Fig. 5-5 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.

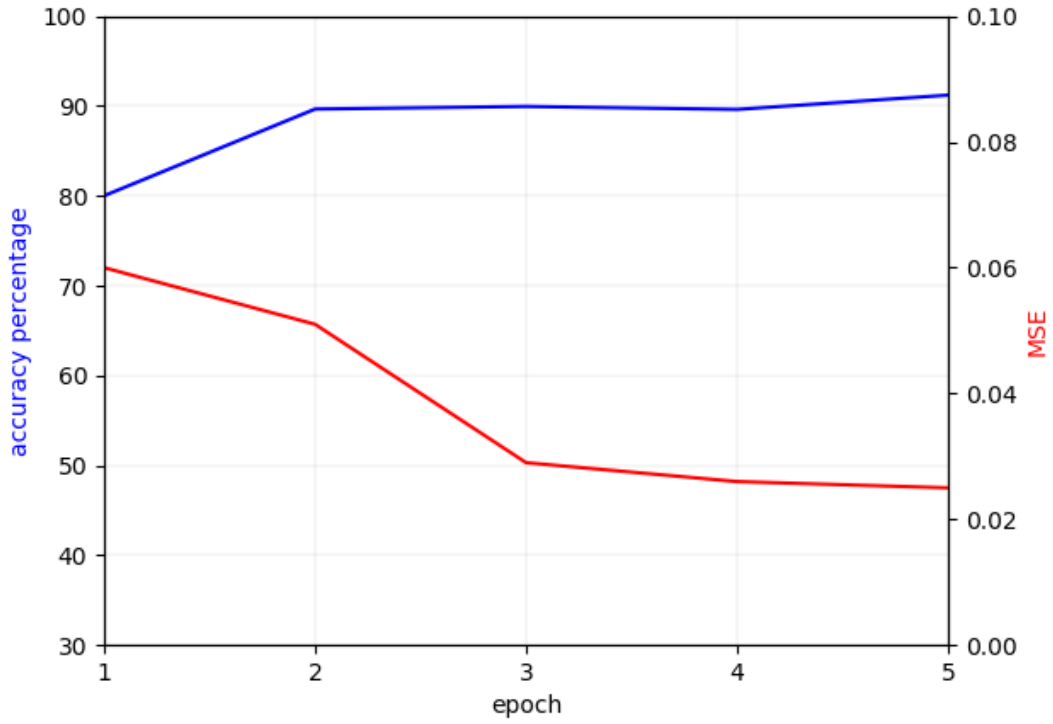

Fig. 5-5. Accuracy and MSE during 5 epochs in the late fusion of the EDLM model in the MIntPAIN database.

Table 5-2 and Fig. 5-6 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 5-2. 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 (%)**	**PR curve (%)**
Average	0.0245	0.0341	92.26	93.67	95
Best	0.02102	0.028	95	95.2	98
Worst	0.03056	0.039	89	91.4	93

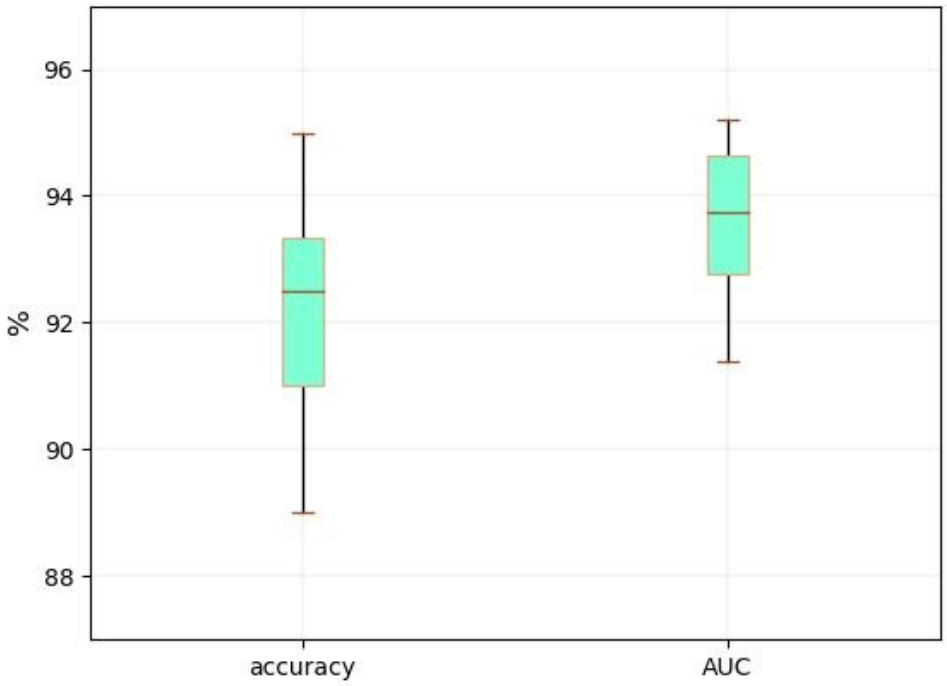

Fig. 5-6. Box plots of Accuracy and AUC for the proposed EDLM model in the MIntPAIN database.

Fig. 5-6 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 shown as yellow, minimum and maximum shown as blue lines. Midian is demonstrated 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 were 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 5-3 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 5-3. 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
PR curves (%)	90	87	88	92	94

The accuracy of the proposed EDLM model was 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.

5.3.2 UNBC-McMaster Shoulder Pain Database's 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 PSPI labels per each frame was used. 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 5-4 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 5-4. The average performance of the proposed model (EDLM) in the UNBC-McMaster Shoulder Pain database for 10-fold cross validation.

MSE	MAE	Accuracy (%)	AUC (%)	PR curves (%)
0.081	0.103	86	90.5	93.5

5.4 Discussion

We compared the obtained results from the EDLM with a baseline model which was designed based on a standard VGG-Face and one stream LSTM model. Table 5-5

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

Classification models	**AUC (%)**	**Accuracy (%)**	**PR curves (%)**
VGGFace + 1 stream LSTM	87	83.4	86
The proposed EDLM model	*93.67*	*92.26*	*95*

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 5-6 shows the learning time of the EDLM for two databases in comparison with two different baseline models. As is indicated in Table 5-6, the total time complexity of the proposed EDLM algorithm for the UNBC-McMaster Shoulder Pain database was 5900 s and the time complexity of it for the MIntPAIN database was 41700 s. As a result, the most time-consuming section of the EDLM was 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 were important factors which affect the complexity and learning time of the algorithm.

The EDLM model demonstrated the highest performance in compare with the other models and the state-of-the-art results. Table 5-7 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 both databases compared with the other research works.

Table 5-6. 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**	**FE Time complexity (based on second and number of applied epochs)**	**classification Time complexity (based on second and number of applied epochs)**	**Sum (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 5-7. Comparing the proposed EDLM with the other state-of-the-art procedures in pain intensity recognition in LOOCV.

Ref	**Classifier**	**Pain level**	**AUC (%)**	**Accuracy (%)**	**MSE**	**Database**	**Data size**
(Lucey, Cohn, Prkachin, et al., 2011)	SVM	2	83.9	-	-	UNBC-McMaster	All
(Lucey, Cohn, Matthews, et al., 2011)	SVM	2	84.7	-	-	UNBC-McMaster	All
(Rodriguez et al., 2017)	CNN-LSTM	2	93.3	83.1	0.74	UNBC-McMaster	Down-up
(Bellantonio et al., 2016)	CNN-RNN	3	-	61.9	-	UNBC-McMaster	Down-up
(Zhou et al., 2016)	-	2	-	-	1.54	UNBC-McMaster	16657 images
(Haque et al., 2018)	CNN-LSTM	5		32.40		MIntPAIN	All
(Bargshady et al., 2020a)	EJH-CNN-BiLSTM	4	87.7	85	0.207	UNBC-McMaster	10783 images
The proposed EDLM	*Ensemble CNN-RNN*	*5*	*91*	*93*	*0.16*	*MIntPAIN*	*34800 images*
The proposed EDLM	*Ensemble CNN-RNN*	*5*	*88*	*90.5*	*0.23*	*UNBC-McMaster*	*10783 images*

5.5 Chapter Summary

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 was 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 were 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 was 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. the deep learning EDLM model **was** found to attain an optimal accuracy evidenced by a relatively lower error compared with the other benchmarked models.

CHAPTER 6

The Proposed HSV-TCN Model

Although, the effectiveness of the proposed models in Chapter 4 and 5 were outstanding, one of main drawbacks of RNNs were the exploding and vanishing gradient problems and the difficulties associated with parallel training and separation tasks. Recent literatures show the feedforward convolution model can be empirically superior to the recurrence models, and can thus be parallelized, making it easier to train the system with a more stable gradient function (Bai et al., 2018).

deep TCN deep learning networks has proven to be an effective method in sequence-based modeling (e.g. a video image). Notably, the TCNs were as an alternative tool to the conventional deep learning (or RNNs) and were used in various classification and modelling tasks. Each layer in the TCN system contains a 1-D convolution block with an increased dilation factor. TCNs can capture the action compositions, segment durations, and long-range dependencies, and were over a magnitude faster to train than competing LSTM-based Recurrent Neural Networks (Bai et al., 2018; Lea et al., 2016). In the facial expression recognition field, the TCN algorithm has high performance. For example, Feng (2019) applied TCNs for measuring stress levels from the face, and Thomas et al. (2018) used TCN in predicting engagement intensity from the facial expressions. In the following the TCN architecture is explained. TCNs were proposed as an alternative tool to RNN algorithms adopted in various classification tasks (Lea et al., 2017; Lea et al., 2016).

To overcome the RNNs challenges and further support potential applications of pain detection technologies in health informatics area, this Chapter aims to develop a Temporal Convolutional Network (TCN) algorithm that can analyze and model the information from video frames in HSV (Hue, Saturation and Value) color space. Based on literature, the choices of color space may have a significant influence on the results of image segmentation. There were many kinds of color space, including RGB, YCbCr, YUV, CIELAB, and HSV (Chen et al., 2008). This study relies on HSV (Smith, 2002), which were shown to yield better results for image segmentation than the RGB color space, were capable of emphasizing human visual perception in hues and have an easily invertible transform from the RGB system (Chen et al., 2007;

Huang & Liu, 2007; Sural et al., 2002). According to Zarit et al. (Zarit et al., 1999), HSV is able to generate the best performance for skin pixel detection. HSV color space gives good results in lighten faces, HSL color space can also yield relatively good results for multi faces and HSI color space gives good results for single faces and zoomed faces videos (Elaw et al., 2019).

we hypothesize that it can be relatively useful to consider the face differences and the subject differences encountered in the detection process and a new TCN classifier with enhance HSV color space input images. The proposed model is named as HSV-TCN includes four key components: image pre-processing, converting images into the enhanced HSV, VGGFace-PCA feature extractor, and TCN pain classifier as shown in Fig. 6-1.

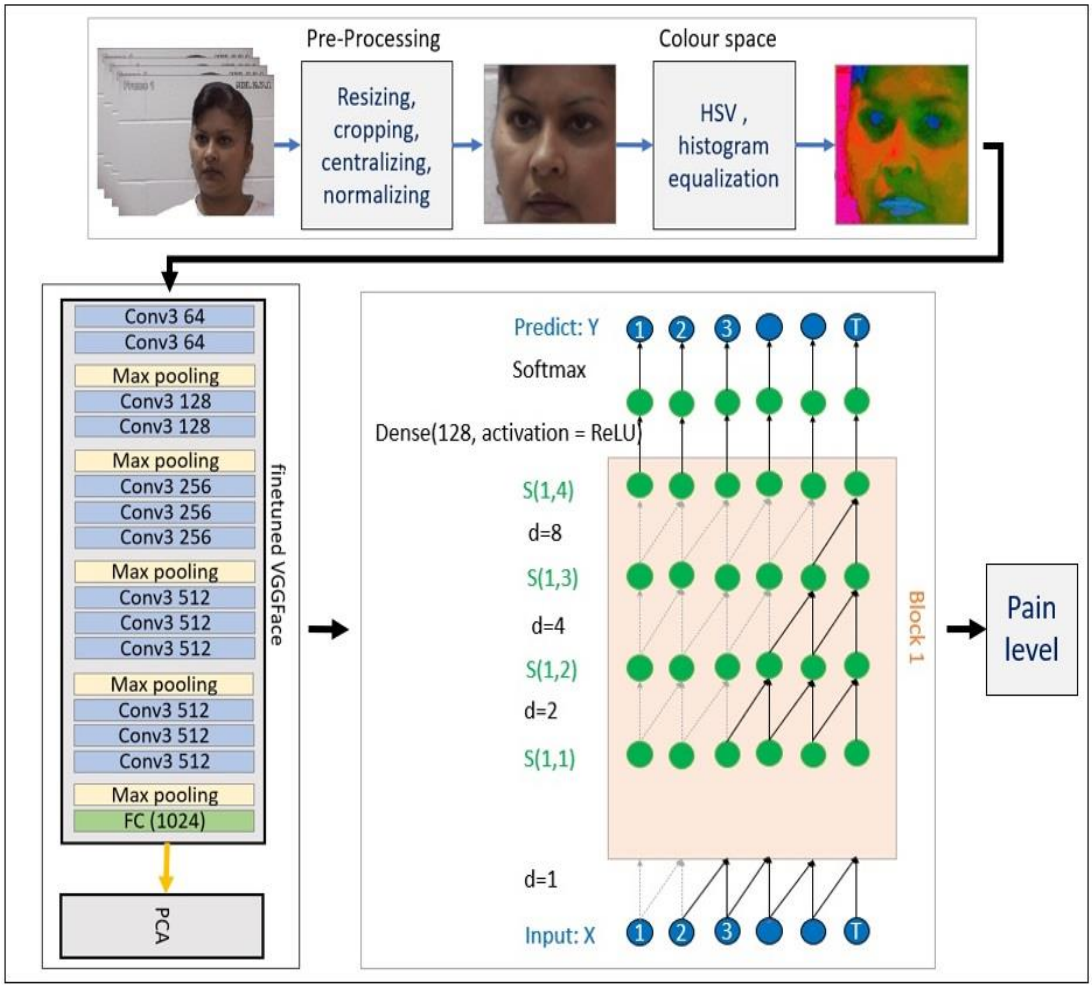

Fig. 6-1. The proposed framework based on the integrated CNN-TCN algorithm with HSV colour space input to implement a facial pain detection system from video frames.

The raw images in this study were first applied in the image preprocessing step utilizing the tasks of centralizing, cropping, and normalization to generate accurate performances and then converted to the HSV color space. Then the converted images have been transferred into our proposed feature extraction VGGFace-PCA introduced in Section 4.2. Next, the extracted features transferred into proposed TCN classifier which has been modified to pain recognition task and has been trained and tested by the UNBC-McMaster Shoulder Pain and MIntPAIN databases.

The results indicate the notion that that HSV images could be more suitable for feature extraction and classification purposes and help in real-time applications that needs to speed up the proposed algorithm. The evaluated results show the proposed TCN classifier with enhance HSV color space input images and VGGFace-PCA feature extractor outperform pain recognition systems from facial videos' images. in terms of the efficiency and complexity its performance was better RNNs models. In the following the details of enhance HSV color converting and the presented TCN classifier with obtained evaluated results and discussion were elaborated. We applied the same image pre-processing and feature extraction techniques introduced in the Chapter 4. in this chapter only the converting to the HSV color space, applied histogram equalization, and applied TCN classifier have been discussed and elaborated.

6.1 Converting RGB to HSV Color Space

In a real-world scenario, an image dataset may be taken in a variety of conditions such as different orientations, location, scales, and brightness. the raw frames were preprocessed by using image processing techniques such as the resizing, face detecting, normalizing, cropping, and centralizing to improve the identification of the images during experimental phase. the images were resized to 224×224×3 pixels because this representation was the most common input size for most of the deep neural network models after cropping including the VGGFace. And then OpenCV face detection and centralizing techniques were applied as described in the Chapter 4.

Then the postprocessed RGB (Swain & Ballard, 1991) video images were converted to an enhanced HSV color space. Based on the literature, choices of color space may have a significant influence on the results of image segmentation. There were many

kinds of color space, including RGB, YCbCr, YUV, and HSV (Chen et al., 2008). HSV (Hue, Saturation, Value) (Smith, 2002), which were shown to yield better results for image segmentation than the RGB color space, and were capable of emphasizing human visual perception in hues and has an easily invertible transform from the RGB system (Chen et al., 2007; Huang & Liu, 2007; Sural et al., 2002). The HSV color space is motivated by the human visual system. In the HSV color space, the luminous component (brightness) is decoupled from color-carrying information (hue and saturation). The transformation of color images in RGB color space into HSV color space is defined as following (Rahman et al., 2014).

- H is a kind of color and the range is 0–360 degrees.
- S is the vividness of color and the range is 0–100%.
- V is the brightness of a color and the range is 0–100%.

$$H = \begin{cases} H_i & \text{if } B \leq G \\ 360 - H_i & \text{if } B > G \end{cases} \tag{6-1}$$

$$H = arc\cos\left(\frac{\frac{1}{2(R-G)+(R-B)}}{[(R-G)^2+(R-B)(G-B)]^{\frac{1}{2}}}\right) \tag{6-2}$$

$$S = \frac{\max(R,G,B)-\min(R,G,B)}{255} \tag{6-3}$$

$$V = \frac{\max(R,G,B)}{255} \tag{6-4}$$

The features extracted in the HSV color space can capture the distinct characteristics of computer graphics better. For example, computer graphics is more color smooth than photographic images in the texture area. Fewer colors were contained in computer graphics. Intensity of computer graphics reveals different characteristic of edge and shade. These differences between computer graphics and photographic images were best described by decoupling the intensity from chromatic information, say, hue and saturation. HSV images were more proper for feature extraction and classification purposes. In the next step to increase the contract of the video frames histogram equalization was applied. The transformation function uniformly spreads out the most frequent intensity values to improve the global contrast.

6.1.1 Histogram Equalizations

Histogram equalization was applied in the next step to increase the contract of the video frames. The transformation function uniformly spreads out the most frequent intensity values to improve the global contrast. Histogram equalization (HE) was applied to increase the contract of the video frames. Histogram equalization is best method for image enhancement. It provides better quality of images without loss of any information. The method of HE is explained as follows (Hitam et al., 2013): The proposed image as $F(i,j)$, with N pixels and gray levels of [0, k-1]. Thus, the probability density function of the image was calculated according to Equation 6-5:

$$\mathrm{P(k)} = \frac{\mathrm{n_k}}{\mathrm{N}} \tag{6-5}$$

Where, n_k is the total number of pixels with the number of grayscale k in the image, the cumulative distribution function (CDF) of the image F (i, j) can be found by the following:

$$\mathrm{C(k)} = \sum_{\mathrm{m=0}}^{\mathrm{k}} \mathrm{P_m} \tag{6-6}$$

Using the CDF values in Equation 6-7, HE matches an input level k to an output level H_k using the level mapping equation:

$$\mathrm{H_k} = \mathrm{(k-1)C(k)} \tag{6-7}$$

Thus, the gain H_k at the output level for the conventional HE that is previously described above can be obtained using Equation 6-8:

$$\Delta \mathrm{H_k} = \mathrm{H_k} - (\mathrm{H_k} - 1) = \mathrm{(k-1)P(k)} \tag{6-8}$$

In other words, the increase in the level of H_k is proportional to the probability of the corresponding level of k in the original image. Theoretically, for images with continuous intensity levels and probability density functions, such a mapping scheme can perfectly equalize the histogram.

In the proposed model the V space of the HSV color space enhanced. to transform HSV space while enhancing only the S space with enhancement factor. The Fig. 6-2 indicates the framework applied in this research work for image processing.

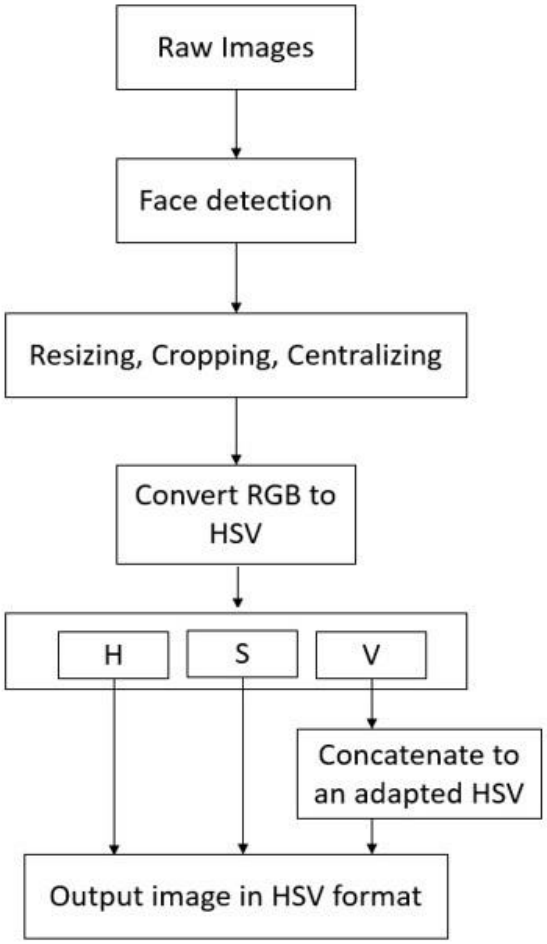

Fig. 6-2. The proposed image processing framework was applied for the raw images.

Fig. 4 shows the process of image processing steps on a sample image of UNBC-McMaster Shoulder Pain database. As it is shown in this figure the image after face detection, centralizing, resizing has been converted into HSV format and then the final step was histogram equalization by V color space. Then, the enhanced HSV color space image format was transferred into the proposed pain detection algorithm.

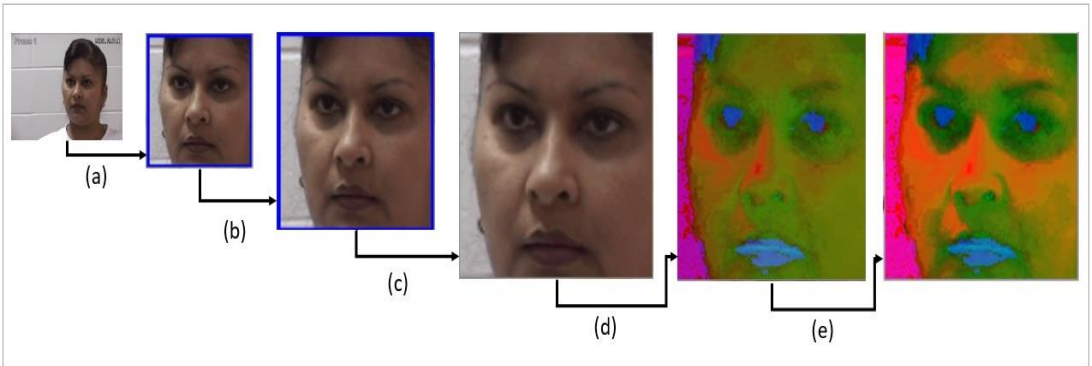

Fig. 6-3. Image processing steps for an image from the UNBC-McMaster Shoulder Pain database. (a) face detection, (b) centralizing, (c) resizing, (d) converting into HSV, and (e) histogram equalization.

6.2 TCN Classifier for Pain Recognition

In this thesis research work, a new TCN based classifier was developed by modifying the TCN to make it suitable for pain detection task from video post-processed images in HSV format.

TCNs were a class of temporal models that use a hierarchy of temporal convolutions to perform the fine-grained action segmentation and subsequent detection of patterns within a dataset (Lea et al., 2017). They can accept variable length inputs like the other sequential models. In the following there were some reasons enable us to use TCNs over recurrent models. TCNs were a class of temporal models that use a hierarchy of temporal convolutions to perform the fine-grained action segmentation and subsequent detection of patterns within a dataset (Lea et al., 2017). They can accept variable length inputs like the other sequential models. In the following there were some reasons enable us to use TCNs over recurrent models.

- TCNs have longer memory in comparison with recurrent neural networks with the same capacity.
- The architecture of the TCNs were parallelizable with flexible receptive field.
- Training TCN models require less memory and time.

TCNs have longer memory in comparison with recurrent neural networks with the same capacity. The architecture of the TCNs were parallelizable with flexible receptive field. Training TCN models require less memory and time. Each layer in the TCN system contains a 1-D convolution block with an increased dilation factor. TCNs can

capture the action compositions, segment durations, and long-range dependencies, and were over a magnitude faster to train than competing LSTM-based Recurrent Neural Networks.

Lea et al. (2017) defined two TCNs including ED-TCN as an encoder-decoder architecture with temporal convolutions and the Dilated TCN, which was adapted from the WaveNet model (Oord et al., 2016), uses a deep series of dilated convolutions. The following id the properties of the TCNs.

1. computations were performed layer-wise, meaning every time-step was updated simultaneously, instead of updating sequentially per-frame.
2. convolutions were computed across time, and predictions at each frame were a function of a fixed-length period, which was referred to as the receptive field.

Fig. 6-4 illustrates the theoretical details of the dilated TCNs architecture taken from (Lea et al., 2017) designed for video analysis the idea was adapted from WaveNet (Oord et al., 2016) is designed for speech analysis.

Suppose $X_t \in R^{F_0}$ is the input feature vector of length F_0 for timestep t for $1 \leq t \leq T$. The number of time steps T may vary for each video sequence. The action label for each frame is given by vector $Y_t \in \{0,1\}^C$ where C is the number of classes.

Y_t is the current action given the video features up to t. Each series of the blocks indicate in Fig. 6-4 contains a sequence of L convolutional layers. The activations in the $l-th$ layer and $j-th$ block is given by $S^{(i,l)} \in R^{F_w \times T}$.

The input into each block $S^{(j,l)}$ is the output from the previous block $S^{(j-1,l)}$, except for the first block which defined as the input data. Each layer has the same number of filters F_w, which enables us to combine activations from different layers using skip connections later. Each layer consists a set of dilated convolutions with rate parameter s, a non-linear activation $f(.)$, and a residual connection than combines the layer's input and the convolution signal.

Convolutions were only applied over two-time steps, t and $t-s$. The filters were parameterized by $w = \{w^{(1)}, w^{(2)}\}$ with $w^{(i)} \in R^{F_w \times F_w}$ and bias vector $b \in R^{F_w}$. $\hat{S}_t^{(j,l)}$ was the result of the dilated convolution at time t. $S_t^{(j,l)}$ was the result after adding the residual connection.

$$\hat{S}_t^{(j,l)} = f(W^{(1)}S_{t-s}^{(j,l-1)} + W^{(2)}S_t^{(j,l-1)} + b) \quad (6\text{-}9)$$

$$S_t^{(j,l)} = S_t^{(j,l-1)} + V\hat{S}_t^{(j,l)} + e \quad (6\text{-}10)$$

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

and

$e \in R^{F_w}$

are a set of weights and biases for the residual and parameters

$\{W, b, V, e\}$

are separate for each layer.

The dilation rate increases for consecutive layers within a block such that $S_l = 2^l$. This enables us to increase the receptive field by a substantial amount without drastically increasing the number of parameters.

The output of each block was summed using a set of skip connections with

$Z^{(0)} \in R^{F_w \times F_w}$

such that

$$Z_t^{(0)} = ReLU(\sum_{j=1}^{B} S_t^{(j,L)}) \quad (6\text{-}11)$$

There is a set of latent states

$$Z_t^{(1)} = ReLU(V_r Z_t^{(0)} + e_r) \quad (6\text{-}12)$$

for weight matrix

$$V_r \in R^{F_w \times F_w} \quad (6\text{-}13)$$

and bias

e_r.

The predictions for each time t were given by

$$\hat{Y}_t = softmax(UZ_t^{(1)} + c). \quad (6\text{-}14)$$

The predictions for each time t were given by

$$U \in R^{C \times F_w} \quad (6\text{-}15)$$

and bias

$c \in R^c$. (6-16)

The filters in each Dilated TCN layer were smaller than in ED-TCN, so to get an equal-sized receptive field it needs more layers or blocks. The receptive field is of length $r(B, L) = B * 2^L$ for number of blocks B and number of layers per block L.

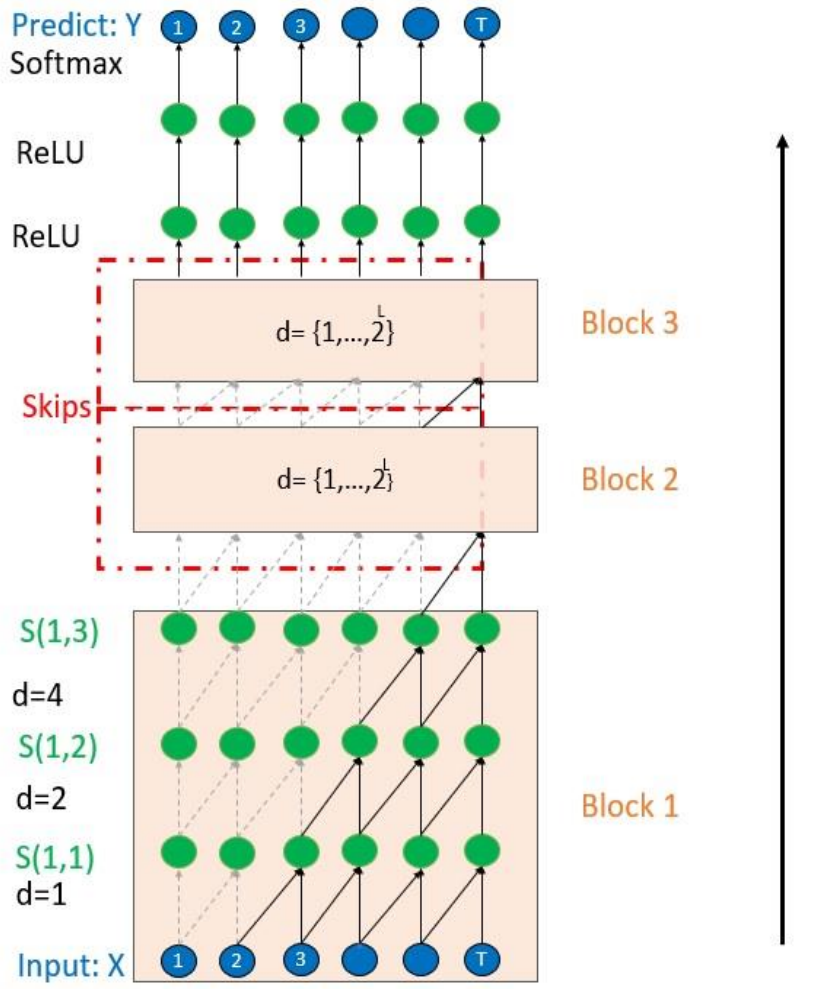

Fig. 6-4. The Dilated TCN model uses a deep stack of dilated convolutions to capture long-range temporal patterns presented by (Lea et al., 2017). The grey dashed lines show the network connections shifted back one-time step. L is convolutional layers, d is dilation rate, the activations in the l-th layer and j-th block were given by $S^{(j,l)} \in R^{F_w \times T}$.

In our proposed model, dilated TCN structure as presented in (Lea et al., 2017) was adapted from original architecture to apply for pain detection task from facial expression video frames. While the number of filters and their shape were quite standard properties of CNNs, the number of filters and layers, the kernel size, their dilation rates, and the number of times the model can be stacked in TCN is highly parameterizable (Lea et al., 2017). in the new proposed model, *d* exponentially was

increased as illustrated in Fig. 6-5. This new developed TCN has four layers of 1-D Conv modules with dilation rates 1, 2, 4, 8 respectively and the input to the TCN is a 4-dimensional feature vector derived from fine-tuned VGGFace-PCA for the UNBC-McMaster Shoulder Pain database and 5-dimentional for the MIntPAIN. For this problem timestep = 4 and batch size = 20 worked well. Then the TCN output layer finetuned by adding an extra fully connected layer in Tensor flow as *Dense (128, activation='ReLU')* and set the output to calculate for *four classes* with activation *Softmax*. The network was optimized with NADAM. It is a kind of Adam optimizer with Nesterov momentum (Dozat, 2016) and has proved more stable than SGD. In training *loss* selected as *categorical_crossentropy* to calculate accuracy.

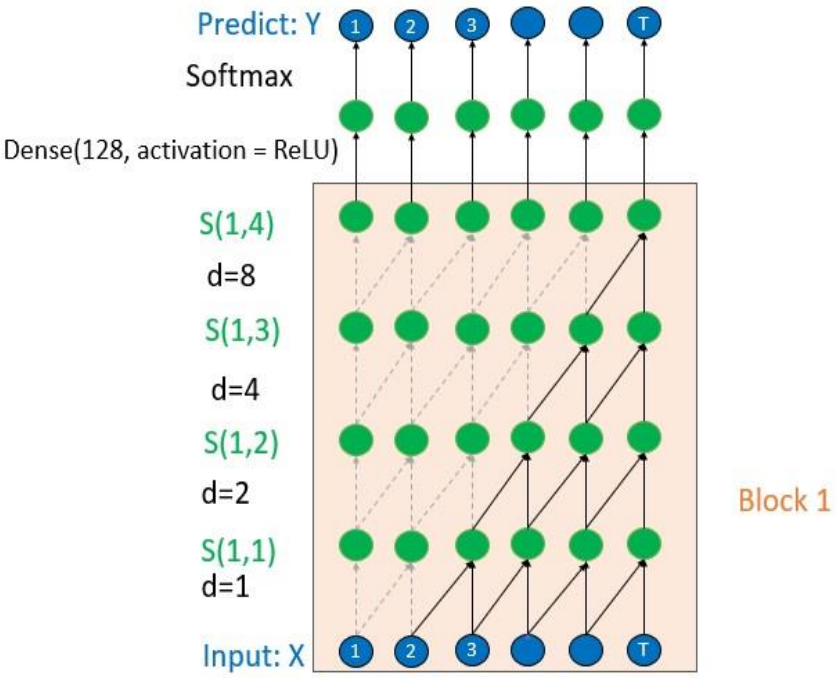

Fig. 6-5. The proposed modified Dilated TCN model used in our pain intensity recognition framework from facial video frames. The grey dashed lines show the network connections shifted back one-time step. d is dilation rate, the activations in the 4 layer and 1 block were given by $S^{(1,4)} \in R^{F_w \times 4}$.

Tables 6-1 and Table 6-2 indicate the details of the TCN parameters used in the proposed model for pain intensity recognition.

Table 6-1. The modified TCN pain detection algorithm modeling parameters from the proposed framework.

TCN parameter	**Parameter size**
Batch size	20
Timestep	4
Dilation	1,2,4,8
Dropout	0.5
Dense	128
Activation	ReLU
Output	4
Activation	Softmax

Table 6-2. The modified TCN pain detection algorithm training parameters from the proposed framework.

Training parameter	Parameter size
Batch size	20
Optimizer	Nadam
Epoch	1
Loss	categorical_crossentropy
Metrics	MAE, MSE, accuracy, AUC

The details of the proposed CNN-TCN model were summarized in Algorithm 6-1. It is noteworthy that this has five epochs with 48 batches to train and test the proposed algorithm.

Algorithm 6-1: HSV-TCN

```
1   Procedure HSV-TCN (input, n, j, batch)
2       Pre-process (input, f)
3       HSV (f, T)
4       Histogram-Equalization (T, M)
5       for k ← 0, n do
6           Finetune-VGGFace (M, features)
7           for epoch ← 0, j do
8               Train-Test (Finetune-VGGFace ())
9           end for
10          TCN (SF, Output)
11          for epoch ← 0, j do
12              Train-Test (TCN ())
13              evaluation (TCN ())
14          end for
15      end for
16  end Procedure
```

6.3 Experimental and Results

In this section the obtained results of the proposed HSV-TCN framework and related plots were explained and illustrated. The proposed algorithm was trained and tested in two databases including the UNBC-McMaster Shoulder Pain database and the MIntPAIN database described in Chapter 3.

6.3.1 UNBC McMaster Shoulder Pain Database's Results

The presented VGGFace-PCA was trained in an epoch to learn of the feature extraction model and reach its best performance. The obtained results show the HSV color space inputs speed up training time of feature extraction algorithms. The proposed TCN classifier was trained and tested by selected features in an epoch as well. Two performance measure were applied including LOOCV and 10-fold-CV. The LOOCV was used on 24 subjects for the training set and one subject for the testing dataset and 25 times it was repeated by replacing different test set. Several performance

evaluations measures, including the classification accuracy, MAE, MSE, AUC were used to evaluate the performance of the proposed model. The results obtained indicated the proposed framework had a high performance in detecting pain in four distinct levels. Table 6-3 shows the average values of the MSE, MAE, accuracy, and AUC of the average for LOSOCV performance measurement.

Table 6-3. The average performance of the proposed HSV-TCN model measured by LOOCV for 25 subjects for four classes in the UNBC-McMaster Shoulder Pain database.

MSE	MAE	Accuracy (%)	AUC (%)	PR curves (%)
0.0692	0.1021	92.44	85	88.5

Fig. 6-6 shows a box plot representing accuracy and AUC performance metrics.

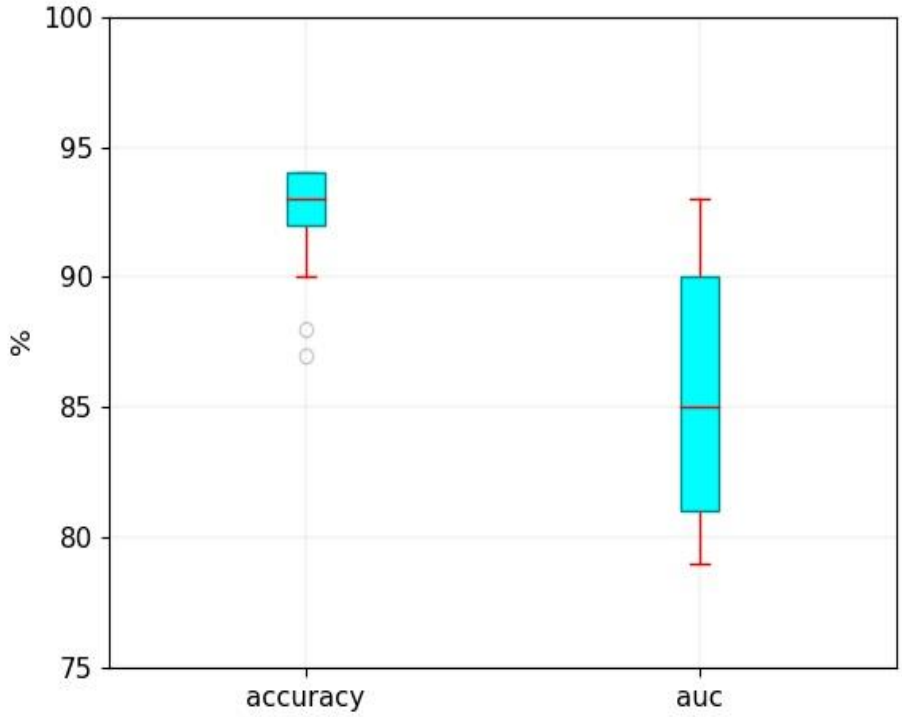

Fig. 6-6. Box plot of the measured performance of the proposed HSV-TCN algorithm includes accuracy, and AUC.

Fig. 6-7 indicates a box plot representing MSE and MAE performance metrics.

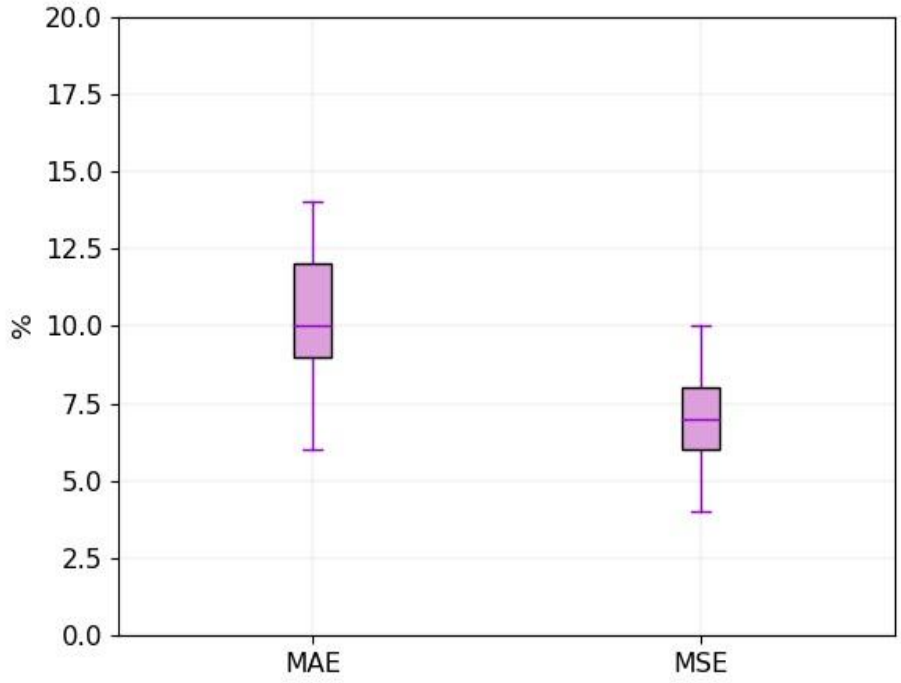

Fig. 6-7. Box plot of the measured performance of the proposed HSV-TCN algorithm includes accuracy, and AUC.

The proposed algorithm then was trained and tested by 10-cross validation technique to compare the results. Table 6-4 shows the obtained results from average accuracy, AUC, MSE, and MAE 10-cross validation.

Table 6-4. The average performance of the proposed HSV-TCN model measured by 10-fold-CV for 25 subjects for four classes in the UNBC-McMaster Shoulder Pain database.

MSE	**MAE**	**Accuracy (%)**	**AUC (%)**	**PR curves (%)**
0.186	0.234	94.14	91.3	93

The performance of the proposed model was measured per each class for the UNBC-McMaster Shoulder Pain database. Table 6-5 indicates measured the results per each class.

Table 6-5. Average pain level per four classes of the proposed HSV-TCN model based on TP, f-score, precision by 10-fold-CV in the UNBC-McMaster Shoulder Pain database.

Class	TP (%)	f-measure (%)	Precision (%)
No pain	89	89.31	88.95
Weak pain	91.2	90.78	91.5
Mild pain	90.30	88.3	90
Strong pain	89.30	90	91.54

This study explored the utility of an LSTM baseline model using the same feature extraction techniques but with the original RGB video frames. The extracted features of the RGB inputs were then passed to an LSTM model. The model consisted of an LSTM layer with a fully connected output layer. Notably, the LSTM cell had 32 hidden units, which were optimized using the ADAM optimizer and considering the MSE and the MAE as the accuracy metric. The network was trained for 1 epoch and implemented in TensorFlow; whose validation results were shown in Table 6-6.

Table 6-6. The average performance of the LSTM model as measured by LOOCV for 25 subjects for four classes on RGB inputs.

MSE	MAE	Accuracy (%)	AUC (%)	PR curves (%)
0.1254	0.198	83	78	80.7

It can be noted that the LSTM baseline model achieved an accuracy of about 83% while the MSE was 0.1254. The proposed Dilated-TCN based model resulted in accuracy about 92.4% with an MSE of only 0.0692. the improvement of the proposed TCN based algorithm on the HSV input was considerably higher than the LSTM model performance with a set of RGB inputs. The AUC of the proposed model, however, was significantly higher than the prescribed LSTM model.

6.3.2 MIntPAIN Database's Results

The proposed HSV-TCN model was trained and evaluated on the MIntPAIN database. Table 6-7 indicates the obtained results of the proposed HSV-TCN model on the

MIntPAIN database measured by accuracy, AUC, MAE, and MSE based on 10-fold cross validation.

Table 6-7. The average performance, best result, and worst results of the proposed HSV-TCN model on the MIntPAIN database for 10-fold cross validation.

Results	**MSE**	**MAE**	**Accuracy (%)**	**AUC (%)**	**PR curves (%)**
Average	0.22	0.26	89	92	94.3

6.4 Discussion

The obtained results compared with the different deep learning algorithms, which recently applied in the same databases to detect pain from the human facial expressions, as indicated in Table 6-8. The comparison results of the proposed HSV-TCN model with the state-of-the-art results show that the proposed framework is significantly effective and efficient. In terms of the error measurements, it has fewer errors measured by MAE and MSE in comparison with results obtained by (Rodriguez et al., 2017). Furthermore, this method was achieved the highest accuracy by 92.44% for LOOCV and 94.14% for 10-fold-CV in the UMBC-McMaster Shoulder in comparison with the results of the other deep learning algorithms.

In comparison with state-of-the-art deep learning pain detection models from facial expressions, the obtained results demonstrated that the proposed HSV-TCN approach can achieve high performance in a multi-classification task. For example, as indicated in Table 6-8 the RNN model proposed by (Bellantonio et al., 2016) achieved less accuracy than our proposed model in the same database and for three classes.

Table 6-8. Comparison of the proposed framework with state-of-the-art results in the UNBC-McMaster Shoulder Pain database in LOOCV.

References	**Pain Level**	**Classifier**	**AUC (%)**	**Accuracy (%)**	**MSE (%)**	**Data Size**
(Lucey, Cohn, Prkachin, et al., 2011)	2	SVM	83.9	-	-	All
(Lucey, Cohn, Matthews, et al., 2011)	2	SVM	84.7	-	-	All
(Rodriguez et al., 2017)	2	LSTM	93.3	83.1	74	Down-up
(Bellantonio et al., 2016)	3	RNN	-	61.9	-	Down-up
(Hammal & Cohn, 2012)	4	SVM	-	80	-	16657 images
(Bargshady et al., 2019)	4	LSTM	82.7	75.2	95	10783 images
(Bargshady et al., 2020a)	4	CNN-RNN	88.7	85	20.7	10783 images
The proposed model	*4*	*TCN*	*85*	*92.44*	*6.92*	*10783 images*

Finally, the performance of the all proposed models in this thesis including HSV-TCN, EJH-CNN-BiLSTM, and EDLM have been compared. The comparison results indicate the effectiveness of the three proposed models was high. In terms of the efficiency the presented HSV-TCN model has higher performance than the EJH-CNN-BiLSTM and the EDLM models in pain detection task. The learning time of the HSV-TCN was less than other two proposed models. Since all the proposed models have been trained and tested by 10-fold cross validation and just in some cases the results have been calculated in LOOCV validation techniques so in Table 6-9 the obtained results of the three proposed models in two databases were compared based on the 10-fold cross validation.

Table 6-9. The comparison results of the three proposed models in this thesis in terms of effectiveness by 10-fold cross validation.

Model	**Database**	**AUC (%)**	**PR curves (%)**	**Accuracy (%)**	**MSE**
EJH-CNN-BiLSTM	UNBC-McMaster	98.4	98	90	0.03
EDLM	UNBC-McMaster	90.5	93	86	0.081
	MIntPAIN	93.67	95	92.26	0.0245
HSV-TCN	UNBC-McMaster	91.3	93.5	94.14	0.186
	MIntPAIN	92	94.3	89	0.22

The feature extraction VGGFace-PCA with HSV color space only was trained in an epoch to reach good performance for the UNBC-McMaster Shoulder Pain database. Whereas with the RGB color space the 5 epochs were applied for the same database. Fig. 6-8 shows the scatter plot of the applied epoch for training VGGFace-PCA in RGB and HSV color space format for the UNBC-McMaster Shoulder Pain and MIntPAIN databases.

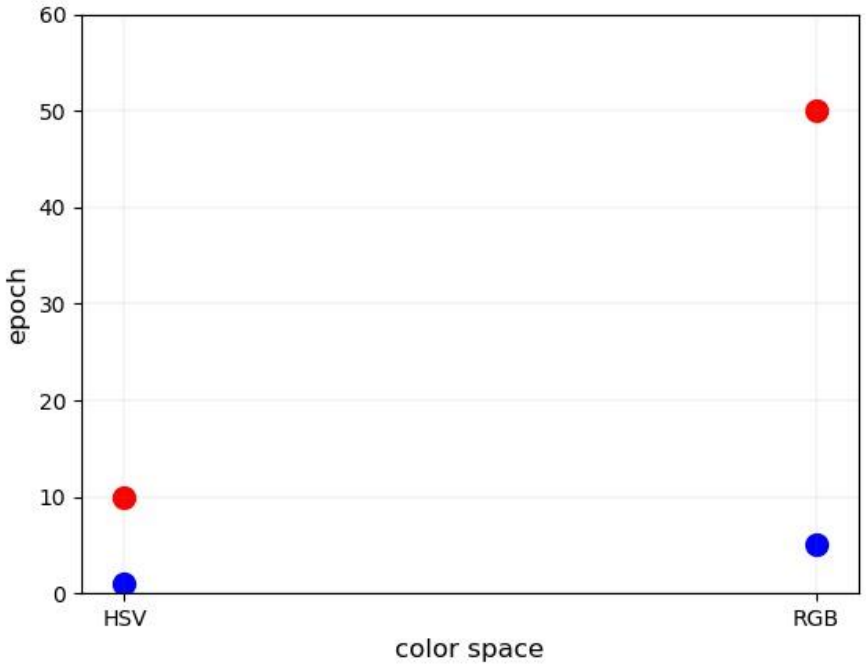

Fig 6-8. The epochs applied for training VGGFace-PCA feature extractor in HSV and RGB colour space. The blue colour indicates the UNBC-McMaster Shoulder Pain database, and the red colour shows the MIntPAIN database.

Table 6-10 indicates the time spent for training the proposed model in both feature extraction and classifier. The comparison results show the learning time for the HSV-TCN was less than other two proposed models in this research work. This thesis research work only focuses on deep learning techniques and for efficiency the obtained results have been compared with deep learning classifiers. It seems that the proposed models based on deep learning in this research work were more efficient than traditional classifiers such as SVM for multi-classes problem. Since SVM for multi-classes needs more calculations such as One-vs-Rest and One-vs-One calculations (Milgram et al., 2006) so its efficiency for the multi-classes may not be better than deep learning algorithms. So, we refer it for the feature works to test SVM effectiveness and efficiency with the selected databases in this research thesis.

Table 6-10. The complexity and speed of the three proposed model in different phase in 10-fold-CV.

Models	Input Image	Classifier	Database	Feature extraction time (s) / epochs	classifier time (s) / epochs
EJH-CNN-BiLSTM (Bargshady et al., 2019)	RGB	BiLSTM	UNBC-McMaster	5300/ 5	560 / 5
EDLM (Bargshady et al., 2020b)	RGB	BiLSTM	UNBC-McMaster	5300/5	600/5
HSV-TCN	HSV	TCN	UNBC-McMaster	1060/1	90/1
EDLM (Bargshady et al., 2020b)	RGB	BiLSTM	MIntPAIN	4000 / 50	1700 / 5
HSV-TCN	HSV	TCN	MIntPAIN	800/10	290/1

Misclassification Rate indicates incorrect predictions and refers to measurements errors. It occurs when an object was assigned to a different class than the one to which they should be assigned (Pham et al., 2019). It is also known as Classification Error. It is calculated by using

$$\text{Misclassification Rate} = \frac{(FP+FN)}{(TP+TN+FP+FN)} \quad (6\text{-}17)$$

or

$$\text{Misclassification Rate} = 1 - \text{Accuracy}. \quad (6\text{-}18)$$

In clinics two types of misclassification bias were of particular importance: *Misclassification of exposure* arises when errors or biases occur during collection of exposure data, and *Misclassification of outcome* derives from errors or biases in the collection of outcome data. They may also influence interpretation of laboratory results or other diagnostic procedures. Regardless of the application, if sensitivity and specificity were less than 100%, some degree of misclassification will occur and may have a profound impact on clinical or research conclusions.

Correcting systematic misclassification errors that occurred during data collection may not be possible when analyzing secondary data sources. care should be taken to minimize the likelihood of misclassification during data collection. This can be accomplished by having a detailed, straightforward, and consistent case definition, strictly following diagnosis guidelines, and minimizing measurement errors by selecting more accurate equipment, tests, or medical examination procedures.

In Chapters 4, 5, and 6 the FP and FN have been calculated per each class and the obtained results show there is small percentage of misclassification occur per each class in three proposed algorithms. Table 6-11 indicates the classification errors of three proposed algorithms in UNBC-McMaster Shoulder Pain and MIntPAIN databases.

Table 6-11. The misclassification error results of the three proposed models by 10-fold cross validation.

Model	Database	Accuracy (%)	1-accuracy (%)
EJH-CNN-BiLSTM	UNBC-McMaster	90	10
EDLM	UNBC-McMaster	86	14
	MIntPAIN	92.26	7.74
HSV-TCN	UNBC-McMaster	94.14	5.86
	MIntPAIN	89	11

If a model after several iterations of cross validation cannot perform to its maximum performance, error analysis could be required to improve the machine learning model performance. If the model does not perform well and the error metric like accuracy is bad, one of the possible solutions is to collect more data. However, collecting more data might take several months that delays the delivery of the project. error analysis should be performed to find out the root cause or causes of bad performance by selecting 100-200 mislabeled samples from the development set (dev set) and do manual error analysis to find out various issues in the mislabeled samples. Table 6-12 shows the example of manual error analysis for images for multi-classes.

Table 6-12. The feature work may like to concentrate more on error analysis.

Samples	Bad performance in class 1	Bad performance in class 2	⋯	Bad performance in class n	Image with multiple classes	Blurry images
Image 1	Yes	No		⋯	No	Yes
Image 2	No	No		⋯	Yes	No
⋮	⋮	⋮		⋮	⋮	⋮
Image 100	Yes	Yes		⋯	No	No

In conclusion, the results obtained from the two databases of the proposed model and comparison with different deep learning models and input space colors indicates that the HSV color space impact the speed of feature extraction process phase and helped us to reach high accuracy in efficient time and less epochs. On the other hand, the obtained results show that the TCN classifier running time was faster than RNN and LDTM deep learning. The accuracy of the TCN classifier also is high. using TCN classifier rather than RNN classifier may be good alternative for automated pain recognition task from facial images. However, the impact of RNNs cannot ignored. the following contributions of this paper were as follows:

- The proposed feature extractor based on finetuned VGG-Face and PCA with HSV color space images has extracted features effectively and in a much timely manner, providing good computational efficiency.
- The proposed and the significantly modified TCN deep learning classifier system was able to recognize the pain level intensity from facial expression video images effectively and efficiently.

The proposed entire framework developed to pain detection system was seen to consume much lesser time in comparison with the RNN deep classifiers with RGB color space inputs and the significant accuracy. it can be applied as an artificial intelligence tool in healthcare software applications to automatically detect pain level from patient faces in a smart, remote, regular, fast, and “anywhere/anytime” manner.

6.5 Chapter Summary

A novel automatic pain intensity recognition algorithm using video images has also been proposed by integrating the CNN and Temporal Convolutional Network algorithms with HSV input color space. The task predicts the four-intensity levels of pain of UNBC-McMaster Shoulder Pain Archive Database 25 patients’ video frames. The proposed framework uses HSV input color space of facial images as the input to a CNN based fine-tuned VGGFace proposed model to extract features and a single Dilated-TCN to classification. The experimental results indicate that there was a considerable improvement from the baseline LSTM model as well as the state-of-the-art deep learning models and the proposed model achieved high performance in terms of accuracy and AUC. Thus, due to its efficiency, the newly proposed algorithm is a good choice as an automate device to monitor the pain levels of individuals faces with

pain. the newly developed CNN-TCN methodology in this paper can use as an artificial intelligence tool, especially as part of an automated pain assessment software applications, including its importance in medical diagnostic areas to support clinicians and other medical researchers in detecting and classifying pain from human facial expressions.

CHAPTER 7

Conclusion and Future Work

In this chapter, conclusion remark including a brief description of the problems, provided answers for the research, contributions have been discussed. Then, limitations of the research work and future work explained.

7.1 Conclusion Remark

Automatic pain management system from facial expressions is a critical tool in measuring pain level and monitoring patients' health conditions. It decreases cost of regular pain measuring by medical staff and improves healthcare systems accurateness in recognizing patients' pain level and plays a crucial role in designing a real-time system that accurately recognizes human facial expressions to support health monitoring and treatment devices. Due to the progress of the data science technology current automatic pain management tools were designed by machine learning based artificial intelligence technology. Machine learning algorithms, implemented as intelligent prediction and classification system, can offer an alternative mechanism for this important task. These algorithms can use information from a camera for collecting the relevant data, to detect both the pain and its relative intensity level. This can be deduced from the movements in facial muscles and its correspondence with the PSPI scores (Prkachin & Solomon, 2008).

Although machine learning and deep learning algorithms have proven as a state-of-the-art diagnostic and prognostic technology, they were not ideal. Their success can be greatly affected by data form, deep learning training and testing structure. Pain detection from facial images continue to suffer from several challenges including the presentation of facial images affected by unbalanced environmental brightness, shooting angle and distance, background interference, and external factors such as the patients smiling during pain or gender-related pain differences. There is also the issue of few facial image databases containing pain labels.

to overcome these challenges and support potential applications of pain detection technology in the health informatics area, and answers the research questions discussed

in the section 1.2 this research developed and evaluated several new deep learning pain recognition systems to find solution to improve pain recognition deep learning algorithms from facial expression. Previous literatures, and current and recent deep learning algorithms in facial expression recognition and pain detection carefully reviewed and considered to answer the RQ 1 as "*What were the most recent deep learning model advancements in pain recognition from facial expressions?*". Based on the reviewed literature discusses in Chapter 2, in computer vision area CNN feature extraction and feature selection or transfer learning methods were performed on data. This research work demonstrates that basic CNNs methods or simple transfer learning feature extraction methods were not well equipped to extracting and selecting important features effectively. As a solution, to answer the RQ 2: "*What is the most effective deep learning algorithm to extract and select features from facial pain images?*", this thesis introduces an efficient and effective algorithm as a modified technique by applying a fine-tuned VGG-Face as a customized pre-trainer and combining its outputs with PCA dimension reduction method to extract and select features from pain image data effectively and efficiently. The proposed method and applied image pre-processing approaches to feed data for this algorithm discusses in Chapter 4 section 4.1 and 4.2. The contribution of this aspect of the research were as follows:

- *Contribution 1: The proposed new feature extraction model composing fine-tuned VGGFace pre-trained and PCA significantly increased the performance of the algorithm in terms of speed and accurate extracted features when compared with the standard VGGFace.*

Three novel enhanced deep learning-based classifiers have been developed and evaluated including the EJH-CNN-BiLSTM, EDML, and HSV-TCN and the obtained results have been compared with the baseline models and other the state-of-the-art models to answer research questions 3 as RQ3: "*What is the most effective and efficient facial expression pain recognition deep learning algorithm to classify pain intensity on multi-level?"*. The comparison results among the proposed models in this research and other recent research in the same databases demonstrated that applying hybrid CNN-BiLSTM and ensemble technique including different hybrid deep learning algorithms has more accurate than a simple deep neural network. The obtained results from the third algorithm as HSV-TCN demonstrated TCN were more efficient than the

BiLSTM for video sequence and training time of the TCN algorithm faster than BiLSTM. The accuracy and effectiveness of the TCN was significant as well. These obtained results and comparison results have been elaborated in Chapter 4,5, and 6 to answer RQ4: "*How effective were the developed enhanced deep learning models to recognize pain from facial expression*?", and RQ5: "*How efficient were the developed enhanced deep learning models to recognize pain from facial expression?".*

In chapter 4 an enhanced joint hybrid deep learning model EJH-CNN-BiLSTM was developed. This model consists of a new feature extraction algorithm connected to new deep learning classifier as a joint and hybrid CNN-BiLSTM classifier was used to classify pain levels. To train and test the algorithm two pin databases which contained facial pain video images with labels including the UNBC-McMaster Shoulder Pain and the MIntPAIN database were utilized. The contributions of this aspect of the research were as follows:

- *Contribution 2: It is concluded that the proposed EJH-CNN-BiLSTM classifier is an effective and accurate model in detecting pain level from facial video images.*

In chapter 5, the proposed model in chapter 4 is extended by applying the hybrid joint CNN-BiLSTM in an ensemble deep learning model. three CNN-BiLSTM models varying in architectures and parameters were designed in a stack ensemble learning model in which their outputs merged into a single model. This proposed model was also trained and tested in the UNBC-McMaster Shoulder Pain and the MIntPAIN databases. By analyzing the results and comparing them with the state-of-the-art results, the following contributions were made:

- *Contribution 3: The proposed ensemble deep learning model, which integrated three independent CNN-RNN deep learners with varying weights and structures, outperformed baseline models and the state-of-the-art methodologies in accuracy.*
- *Contribution 4: The proposed EDLM model is the optimum deep learning method resulting in a low qualified error compared with the other target models.*

In chapter 6, a new deep learning model based on convolutional neural networks and temporal neural networks was designed. Since, RNNs has exploding and vanishing gradient problems and the difficulties associated with parallel training and separation tasks, different types of deep learning models were applied for this algorithm. The

TCNs is a feedforward convolution model which can be superior to the RNNs. a deep dilated TCN algorithm is modified for this task. TCNs were previously applied in different tasks successfully but never applied for pain recognition tasks. In this experiment the input color space was also changed to the HSV due to its effectiveness in facial expression recognition and transferred into the same feature extraction proposed in chapter 4 as VGGFace-PCA. The contribution of the proposed TCNs pain recognition model is:

- *Contribution 5: The proposed TCN algorithm advances automated system design and applications for the health informatics area.*
- *Contribution 6: The experimental results indicate that there is a considerable improvement from the baseline LSTM model and the state-of-the-art deep learning models and the proposed model achieved high performance in terms of accuracy and complexity.*
- *Contribution 7: The newly proposed algorithm is a good choice as an automate device to monitor the pain levels of individual faces since its learning time is faster when compared to other deep learning algorithms.*

In conclusion, the newly developed deep learning methodologies discussed in this thesis can be used as an artificial intelligence tool, especially as part of an automated pain assessment software applications, including its importance in medical diagnostic areas to support clinicians and other medical researchers in detecting and classifying pain from human facial expressions. applying automated pain detection tools can improve clinicians' proficiency by reducing time and cost and increasing accuracy and safety in collecting and analyzing patients' data in both diagnosis steps and treatment procedures. AI pain detection tools could improve patients' symptoms diagnosis and facilitate their recovery progress which is an important achievement for healthcare systems.

7.2 Current Limitations

There were some limitations in this project and in automatic pain detection systems from facial expression development and evaluation which were listed in this section including:

1. In this research PhD thesis, new algorithms developed and evaluated to pain recognition systems only from facial expression video images. The vocal, body movement, and other patients' behaviors data were not considered in this research.
2. This research project focused on enhancing deep learning models although the obtained results were compared with the most important literature review results, other machine learning techniques improvements were not discussed in this research work.
3. In this research two popular and bid databases including UNBC-McMaster Shoulder Pain database and MIntPAIN database used to train and evaluate the presented algorithms. However, one of the challenges was that most of the research into facial expressions, especially in facial pain detection, currently lacks a standard database. Accessing the patients' original images was not possible in the most pain databases. Most of database labels were not complete and missed labeling at the frames levels. This made it challenging for modelling of pain recognition.

7.3 Suggestions for Future Work

This thesis has resolved several challenges regarding the use of automatic pain recognition systems from facial expressions. However, extensive future investigations were still required, including the following:

1. Future work can use different frameworks for pain recognition such as techniques introduced in Zhang, Wang et al. (2017) which firstly recognize the general facial expression, then if it is pain, use fine-grained pain level classification. Deep metric learning methods, such as Siamese networks, may also be used to achieve better performance (Liu et al., 2017).
2. Future work may also consider the loss functions method that performs well on imbalanced datasets (Bi & Zhang, 2018; Lin et al., 2017; Zhang, Bi, et al., 2017; Zhang, Bi, et al., 2019; Zhang, Liu, et al., 2017).
3. The promising results achieved with the newly developed TCN method in this study indicate significant potential for practical applications in future works, including the developing of a real-time approach using a TCN; and exploring the TCNs for other facial expression tasks such as emotion detection.

4. Future study may advance this algorithm in different types of pain face images and video frame databases to further accelerate the speed and accuracy of feature extracting of images for broader real-time applications in health informatics and medical diagnosis areas.
5. In the future, a comprehensive algorithm may be designed to train and test algorithms in automatic pain recognition systems based on all pain behaviors such as facial expressions, vocalizations, body movements and physiological responses. This may complement current assessment methods to achieve better pain management. More knowledge about factors influencing pain would help automatic recognition, as it may allow to better leverage context information. Currently, most of the databases only provide pain image information without additional knowledge about the patients' background.
6. As discussed in the limitation section, one of the big challenges in pain recognition from face data is the limited number of databases that provide access to real patients' pain information and video frames. Future work should be validated on multiple datasets to show consistent performance across diverse data and how well a system generalizes to other conditions, medical populations, and pain types.
7. The FACS and PSPI labelling systems were the most valid and popular pain labeling system for facial expression. Feature studies may consider new facial expression coding systems.
8. In future work, we expect to test and compare the current presented research with a color constancy algorithm.

List of References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., & Isard, M. (2016). Tensorflow: A system for large-scale machine learning. 12th Symposium on Operating Systems Design and Implementation Savannah, GA.

Ahonen, T., Hadid, A., & Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence, 28*(12), 2037-2041.

Akcay, S., Kundegorski, M. E., Willcocks, C. G., & Breckon, T. P. (2018). Using deep convolutional neural network architectures for object classification and detection within X-ray baggage security imagery. *IEEE transactions on information forensics and security, 13*(9), 2203-2215. https://doi.org/10.1109/TIFS.2018.2812196

Ashraf, A. B., Lucey, S., Cohn, J. F., Chen, T., Ambadar, Z., Prkachin, K. M., & Solomon, P. E. (2009). The painful face–pain expression recognition using active appearance models. *Image and vision computing, 27*(12), 1788-1796. https://doi.org/https://doi.org/10.1016/j.imavis.2009.05.007

Athiwaratkun, B., & Kang, K. (2015). Feature representation in convolutional neural networks. *arXiv preprint arXiv:1507.02313*.

Aung, M. S., Kaltwang, S., Romera-Paredes, B., Martinez, B., Singh, A., Cella, M., Valstar, M., Meng, H., Kemp, A., & Shafizadeh, M. (2015). The automatic detection of chronic pain-related expression: requirements, challenges and the multimodal EmoPain dataset. *IEEE Transactions on Affective Computing, 7*(4), 435-451. https://doi.org/10.1109/TAFFC.2015.2462830

Aydede, M. (2017). Defending the IASP definition of pain. *The Monist, 100*(4), 439-464. https://doi.org/ https://doi.org/10.1093/monist/onx021

Bai, S., Kolter, J. Z., & Koltun, V. (2018). An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv*. https://doi.org/preprint arxiv:1803.01271

Bargshady, G., Soar, J., Zhou, X., Deo, R. C., Whittaker, F., & Wang, H. (2019). A joint deep neural network model for pain recognition from face. Proceedings of the 4th IEEE International Conference on Computer and Communication Systems (ICCS 2019), Singapore.

Bargshady, G., Zhou, X., Deo, R. C., Soar, J., Whittaker, F., & Wang, H. (2020a). Enhanced deep learning algorithm development to detect pain intensity from facial expression images. *Expert Systems with Applications, 149*(113305). https://doi.org/https://doi.org/10.1016/j.eswa.2020.113305

Bargshady, G., Zhou, X., Deo, R. C., Soar, J., Whittaker, F., & Wang, H. (2020b). Ensemble neural network approach detecting pain intensity from facial expressions. *Artificial intelligence in medicine*, 101954.

Bartlett, M. S., Littlewort, G. C., Frank, M. G., & Lee, K. (2014). Automatic decoding of facial movements reveals deceptive pain expressions. *Current Biology, 24*(7), 738-743. https://doi.org/https://doi.org/10.1016/j.cub.2014.02.009

Bellantonio, M., Haque, M. A., Rodriguez, P., Nasrollahi, K., Telve, T., Escalera, S., Gonzalez, J., Moeslund, T. B., Rasti, P., & Anbarjafari, G. (2016). Spatio-temporal pain recognition in cnn-based super-resolved facial images. Video Analytics. Face and Facial Expression Recognition and Audience Measurement, Cancun, Mexico.

Bengio, Y., & Grandvalet, Y. (2004). No unbiased estimator of the variance of k-fold cross-validation. *Journal of machine learning research, 5*(Sep), 1089-1105.

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. Proceedings of the eleventh annual conference on Computational learning theory,

Boyd, K., Eng, K. H., & Page, C. D. (2013). Area under the precision-recall curve: point estimates and confidence intervals. Joint European conference on machine learning and knowledge discovery in databases,

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern recognition, 30*(7), 1145-1159. https://doi.org/10.1016/S0031-3203(96)00142-2

Brahnam, S., Chuang, C.-F., Sexton, R. S., & Shih, F. Y. (2007). Machine assessment of neonatal facial expressions of acute pain. *Decision Support Systems, 43*(4), 1242-1254. https://doi.org/https://doi.org/https://doi.org/10.1016/j.dss.2006.02.004

Bruce, C. S. (1994). Research students' early experiences of the dissertation literature review. *Studies in Higher Education, 19*(2), 217-229. https://doi.org/https://doi.org/10.1080/03075079412331382057

Casti, P., Mencattini, A., Comes, M. C., Callari, G., Di Giuseppe, D., Natoli, S., Dauri, M., Daprati, E., & Martinelli, E. (2019). Calibration of Vision-Based Measurement of Pain Intensity With Multiple Expert Observers. *IEEE Transactions on Instrumentation and Measurement, 68*(7), 2442-2450. https://doi.org/10.1109/TIM.2019.2909603

Chen, T.-W., Chen, Y.-L., & Chien, S.-Y. (2008). Fast image segmentation based on K-Means clustering with histograms in HSV color space. 2008 IEEE 10th Workshop on Multimedia Signal Processing,

Chen, W., Shi, Y. Q., & Xuan, G. (2007). Identifying computer graphics using HSV color model and statistical moments of characteristic functions. 2007 ieee international conference on multimedia and expo,

Chen, Z., Ansari, R., & Wilkie, D. (2019). Learning Pain from Action Unit Combinations: A Weakly Supervised Approach via Multiple Instance Learning. *IEEE Transactions on Affective Computing*. https://doi.org/10.1109/TAFFC.2019.2949314

Chibelushi, C. C., & Bourel, F. (2003). Facial expression recognition: A brief tutorial overview. *CVonline: On-Line Compendium of Computer Vision, 9*, 11.

Cootes, T. F., Edwards, G. J., & Taylor, C. J. (1998). Active appearance models. European conference on computer vision,

Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J. (1995). Active shape models-their training and application. *Computer vision and image understanding, 61*(1), 38-59.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05),

Damale, R. C., & Pathak, B. V. (2018, 14-15 June 2018). Face Recognition Based Attendance System Using Machine Learning Algorithms. 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India

Davis, J., & Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. Proceedings of the 23rd international conference on Machine learning,

Dietterich, T. G. (2000). Ensemble methods in machine learning. International workshop on multiple classifier systems, Günzburg, Germany

Ding, C., & Tao, D. (2017). Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE transactions on pattern analysis and machine intelligence, 40*(4), 1002-1014. https://doi.org/10.1109/TPAMI.2017.2700390

Dozat, T. (2016). Incorporating nesterov momentum into adam.

Dyer, C., Ballesteros, M., Ling, W., Matthews, A., & Smith, N. A. (2015). Transition-based dependency parsing with stack long short-term memory. ACL 2015, Beijing.

Edwards, G. J., Taylor, C. J., & Cootes, T. F. (1998). Interpreting face images using active appearance models. Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition,

Egede, J., Valstar, M., & Martinez, B. (2017). Fusing deep learned and hand-crafted features of appearance, shape, and dynamics for automatic pain estimation. 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), Washington DC.

Egede, J., Valstar, M., Torres, M. T., & Sharkey, D. (2019). Automatic Neonatal Pain Estimation: An Acute Pain in Neonates Database. 2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII),

Ekman, P., & Friesen, W. V. (1978). *Facial Action Coding System: Investigator's Guide*. Consulting Psychologists Press. https://books.google.com.au/books?id=7pqFtQAACAAJ

Elaw, S., Abd-Elhafiez, W. M., & Heshmat, M. (2019). Comparison of Video Face Detection methods Using HSV, HSL and HSI Color Spaces. 2019 14th International Conference on Computer Engineering and Systems (ICCES),

Emami, S., & Suciu, V. P. (2012). Facial recognition using OpenCV. *Journal of Mobile, Embedded and Distributed Systems, 4*(1), 38-43.

Feng, S. (2019). Dynamic Facial Stress Recognition in Temporal Convolutional Network. International Conference on Neural Information Processing,

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences, 55*(1), 119-139. https://doi.org/https://doi.org/10.1007/3-540-59119-2_166

Gable, G. G. (1994). Integrating case study and survey research methods: an example in information systems. *European journal of information systems, 3*(2), 112-126. https://doi.org/https://doi.org/10.1057/ejis.1994.12

Gers, F. A., & Schmidhuber, E. (2001). LSTM recurrent networks learn simple context-free and context-sensitive languages. *IEEE Transactions on Neural Networks, 12*(6), 1333-1340. https://doi.org/10.1109/72.963769

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Gulli, A., & Pal, S. (2017). *Deep Learning with Keras*. Packt Publishing Ltd.

Hammal, Z., & Cohn, J. F. (2012). Automatic detection of pain intensity. Proceedings of the 14th ACM international conference on Multimodal interaction, Santa Monica, California, USA.

Han, D., Liu, Q., & Fan, W. (2018). A new image classification method using CNN transfer learning and web data augmentation. *Expert Systems with Applications, 95*, 43-56. https://doi.org/https://doi.org/10.1016/j.eswa.2017.11.028

Han, J., Chen, H., Liu, N., Yan, C., & Li, X. (2017). CNNs-based RGB-D saliency detection via cross-view transfer and multiview fusion. *IEEE transactions on cybernetics, 48*(11), 3171-3183. https://doi.org/10.1109/TCYB.2017.2761775

Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis & Machine Intelligence*(10), 993-1001. https://doi.org/10.1109/34.58871

Haque, M. A., Bautista, R. B., Noroozi, F., Kulkarni, K., Laursen, C. B., Irani, R., Bellantonio, M., Escalera, S., Anbarjafari, G., & Nasrollahi, K. (2018). Deep multimodal pain recognition: a database and comparison of spatio-temporal visual modalities. 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), Xian, China.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition,

Herr, K., Coyne, P. J., McCaffery, M., Manworren, R., & Merkel, S. (2011). Pain assessment in the patient unable to self-report: position statement with clinical practice recommendations. *Pain Management Nursing, 12*(4), 230-250.

Hitam, M. S., Awalludin, E. A., Yussof, W. N. J. H. W., & Bachok, Z. (2013). Mixture contrast limited adaptive histogram equalization for underwater image enhancement. 2013 International conference on computer applications technology (ICCAT),

Howse, J., Joshi, P., & Beyeler, M. (2016). *Opencv: computer vision projects with python*. Packt Publishing Ltd.

Hu, F., Xia, G.-S., Hu, J., & Zhang, L. (2015). Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing, 7*(11), 14680-14707. https://doi.org/https://doi.org/10.3390/rs71114680

Huang, Z.-K., & Liu, D.-H. (2007). Segmentation of color image using EM algorithm in HSV color space. 2007 International Conference on Information Acquisition,

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in science & engineering, 9*(3), 90. https://doi.org/10.1109/MCSE.2007.55

Jeong, D. I., & Kim, Y.-O. (2009). Combining single-value streamflow forecasts–A review and guidelines for selecting techniques. *Journal of Hydrology, 377*(3-4), 284-299. https://doi.org/Show more https://doi.org/10.1016/j.jhydrol.2009.08.028

Kaltwang, S., Rudovic, O., & Pantic, M. (2012). Continuous pain intensity estimation from facial expressions. International Symposium on Visual Computing, Berlin, Heidelberg.

Kaya, A., Keceli, A. S., Catal, C., Yalic, H. Y., Temucin, H., & Tekinerdogan, B. (2019). Analysis of transfer learning for deep neural network based plant classification models. *Computers and electronics in agriculture, 158*, 20-29. https://doi.org/https://doi.org/10.1016/j.compag.2019.01.041

Ketkar, N. (2017). Introduction to keras. In *Deep Learning with Python* (pp. 97-111). Springer. https://doi.org/https://doi.org/10.1007/978-1-4842-2766-4_7

Khan, R. A., Meyer, A., Konik, H., & Bouakaz, S. (2013, 15-19 July 2013). Pain detection through shape and appearance features. 2013 IEEE International Conference on Multimedia and Expo (ICME), San Jose, USA.

Kharghanian, R., Peiravi, A., & Moradi, F. (2016, 16-20 Aug. 2016). Pain detection from facial images using unsupervised feature learning approach. 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Orlando, FL, USA.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, Nevada, USA.

Kuo, C.-M., Lai, S.-H., & Sarkis, M. (2018). A compact deep learning model for robust facial expression recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops,

Lea, C., Flynn, M. D., Vidal, R., Reiter, A., & Hager, G. D. (2017). Temporal convolutional networks for action segmentation and detection. proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,

Lea, C., Vidal, R., Reiter, A., & Hager, G. D. (2016). Temporal convolutional networks: A unified approach to action segmentation. European Conference on Computer Vision,

Leedy, P. D., & Ormrod, J. E. (2005). *Practical research*. Pearson Custom.

Lewis, D. D. (1995). A Sequential Algorithm for Training Text Classifiers: Corrigendum and additional data. *Special Interest Group on Information Retrieval*.

Li, J., Zhao, B., Zhang, H., & Jiao, J. (2009). Face recognition system using svm classifier and feature extraction by pca and lda combination. 2009 International Conference on Computational Intelligence and Software Engineering, Wuhan, China.

Littlewort, G. C., Bartlett, M. S., & Lee, K. (2009). Automatic coding of facial expressions displayed during posed and genuine pain. *Image and vision computing, 27*(12), 1797-1803. https://doi.org/Show more https://doi.org/10.1016/j.imavis.2008.12.010

Liu, D., Cheng, D., Houle, T. T., Chen, L., Zhang, W., & Deng, H. (2018). Machine learning methods for automatic pain assessment using facial expression information: Protocol for a systematic review and meta-analysis. *Medicine, 97*(49).

Liu, X., Vijaya Kumar, B., You, J., & Jia, P. (2017). Adaptive deep metric learning for identity-aware facial expression recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops,

Liu, Y., Peng, Y., Lim, K., & Ling, N. (2019). A novel image retrieval algorithm based on transfer learning and fusion features. *World Wide Web, 22*(3), 1313-1324.

Lucey, P., Cohn, J., Lucey, S., Matthews, I., Sridharan, S., & Prkachin, K. M. (2009). Automatically detecting pain using facial actions. 2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops,

Lucey, P., Cohn, J., Lucey, S., Sridharan, S., & Prkachin, K. M. (2009). Automatically detecting action units from faces of pain: Comparing shape and appearance features. 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops,

Lucey, P., Cohn, J. F., Matthews, I., Lucey, S., Sridharan, S., Howlett, J., & Prkachin, K. M. (2011). Automatically Detecting Pain in Video Through Facial Action Units. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 41*, 664-674.

Lucey, P., Cohn, J. F., Prkachin, K. M., Solomon, P. E., & Matthews, I. (2011). Painful data: The UNBC-McMaster shoulder pain expression archive database. Face and Gesture 2011, Santa Barbara, CA, USA

Lynch, M. E. (2011). The need for a Canadian pain strategy. *Pain Research and Management, 16*(2), 77-80.

Lyons, M., Akamatsu, S., Kamachi, M., & Gyoba, J. (1998). Coding facial expressions with gabor wavelets. Proceedings Third IEEE international conference on automatic face and gesture recognition,

Ma, J., & Yuan, Y. (2019). Dimension reduction of image deep feature using PCA. *Journal of Visual Communication and Image Representation, 63*, 102578.

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

Matuszewski, B. J., Quan, W., Shark, L.-K., Mcloughlin, A. S., Lightbody, C. E., Emsley, H. C., & Watkins, C. L. (2012). Hi4D-ADSIP 3-D dynamic facial articulation database. *Image and vision computing, 30*(10), 713-727.

McKinney, W. (2012). *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. " O'Reilly Media, Inc.".

Milgram, J., Cheriet, M., & Sabourin, R. (2006). “One against one” or “one against all”: Which one is better for handwriting recognition with SVMs?

Minetto, R., Segundo, M. P., & Sarkar, S. (2019). Hydra: an ensemble of convolutional neural networks for geospatial land classification. *IEEE Transactions on Geoscience and Remote Sensing, 57*(9), 6530 - 6541.

Mousavi, R., & Eftekhari, M. (2015). A new ensemble learning methodology based on hybridization of classifier ensemble selection approaches. *Applied Soft Computing, 37*, 652-666.

Mozaffari, S., Behravan, H., & Akbari, R. (2010). Gender classification using single frontal image per person: combination of appearance and geometric based features. 2010 20th International Conference on Pattern Recognition,

Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., & Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering, 22*(10), 1345-1359.

Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep face recognition. bmvc, Swansea, UK.

Payne, R. (2000). Chronic pain: challenges in the assessment and management of cancer pain. *Journal of pain and symptom management, 19*(1), 12-15.

Pedersen, H. (2015). Learning appearance features for pain detection using the UNBC-McMaster shoulder pain expression archive database. International Conference on Computer Vision Systems,

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., & Dubourg, V. (2011). Scikit-learn: Machine learning in Python. *Journal of machine learning research, 12*(Oct), 2825-2830.

Pham, A., Cummings, M., Lindeman, C., Drummond, N., & Williamson, T. (2019). Recognizing misclassification bias in research and medical practice. *Family practice, 36*(6), 804-807.

Phillips, B. S. (1966). *Social research: Strategy and tactics* (Vol. 966). Macmillan New York.

Pitaloka, D. A., Wulandari, A., Basaruddin, T., & Liliana, D. Y. (2017). Enhancing CNN with preprocessing stage in automatic emotion recognition. *Procedia Computer Science, 116*, 523-529.

Powers, D. M. (2011). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, *2*(1), 37-63.

Prkachin, K. M., & Solomon, P. E. (2008). The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain. *Pain, 139*(2), 267-274.

Rahman, M. A., Purnama, I. K. E., & Purnomo, M. H. (2014). Simple method of human skin detection using HSV and YCbCr color spaces. 2014 International Conference on Intelligent Autonomous Agents, Networks and Systems,

Rathee, N., & Ganotra, D. (2015). A novel approach for pain intensity detection based on facial feature deformations. *Journal of Visual Communication and Image Representation, 33*, 247-254.

Rodriguez, P., Cucurull, G., Gonzàlez, J., Gonfaus, J. M., Nasrollahi, K., Moeslund, T. B., & Roca, F. X. (2017). Deep pain: Exploiting long short-term memory networks for facial expression classification. *IEEE transactions on cybernetics*(99), 1-11.

Rudovic, O., Pavlovic, V., & Pantic, M. (2013). Automatic pain intensity estimation with heteroscedastic conditional ordinal random fields. International Symposium on Visual Computing,

Salekin, M. S., Zamzmi, G., Goldgof, D., Kasturi, R., Ho, T., & Sun, Y. (2019). Multi-Channel Neural Network for Assessing Neonatal Pain from Videos. 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC),

Sanner, M. F. (1999). Python: a programming language for software integration and development. *J Mol Graph Model*, *17*(1), 57-61.

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

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*, 85-117.

Selltiz, C., Wrightsman, L. S., & Cook, S. W. (1976). *Research methods in social relations*. Holt, Rinehart and Winston.

Sharif Razavian, A., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). CNN features off-the-shelf: an astounding baseline for recognition. Proceedings of the IEEE conference on computer vision and pattern recognition workshops,

SHARKEY, A. J. C. (1996). On combining artificial neural nets. *Connection Science, 8*(3-4), 299-314.

Simonyan, K., & Zisserman, A. (2015). VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION. ICLR, San Diego.

Smith, J. R. (2002). 11 Color for Image Retrieval. *Image Databases*, 285.

Soar, J., Bargshady, G., Zhou, X., & Whittaker, F. (2018). Deep learning model for detection of pain intensity from facial expression. International Conference on Smart Homes and Health Telematics,

Sun, Y., Chen, Y., Wang, X., & Tang, X. (2014). Deep learning face representation by joint identification-verification. the 27th International Conference on Neural Information Processing Systems Montreal, Canada.

Sun, Y., Wang, X., & Tang, X. (2015). Deeply learned face representations were sparse, selective, and robust. Proceedings of the IEEE conference on computer vision and pattern recognition, Boston, Massachusetts.

Sural, S., Qian, G., & Pramanik, S. (2002). Segmentation and histogram generation using the HSV color space for image retrieval. Proceedings. International Conference on Image Processing,

Susman, G. I. (1983). Action research: a sociotechnical systems perspective. *Beyond method: Strategies for social research, 95*, 113.

Swain, M. J., & Ballard, D. H. (1991). Color indexing. *International Journal of Computer Vision, 7*(1), 11-32.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition,

Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. Proceedings of the IEEE conference on computer vision and pattern recognition, Columbus, Ohio.

Tavakolian, M., & Hadid, A. (2019). A spatiotemporal convolutional neural network for automatic pain intensity estimation from facial dynamics. *International Journal of Computer Vision, 127*(10), 1413-1425.

Theagarajan, R., Bir, B., Angeles, D., & Pala, F. (2018). [Regular Paper] KnowPain: Automated System for Detecting Pain in Neonates from Videos. 2018 IEEE 18th International Conference on Bioinformatics and Bioengineering (BIBE),

Thomas, C., Nair, N., & Jayagopi, D. B. (2018). Predicting Engagement Intensity in the Wild Using Temporal Convolutional Network. Proceedings of the 2018 on International Conference on Multimodal Interaction,

Ueda, T., & Hoshiai, Y. (1997). Application of principal component analysis for parsimonious summarization of DEA inputs and/or outputs. *Journal of the Operations Research Society of Japan, 40*(4), 466-478.

Vehtari, A., Gelman, A., & Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and computing, 27*(5), 1413-1432.

Walecki, R., Rudovic, O., Pavlovic, V., Schuller, B., & Pantic, M. (2017). Deep structured learning for facial expression intensity estimation. *IMAVIS (article in press), 259*, 143-154.

Walter, S., Gruss, S., Ehleiter, H., Tan, J., Traue, H. C., Werner, P., Al-Hamadi, A., Crawcour, S., Andrade, A. O., & da Silva, G. M. (2013). The biovid heat pain database data for the advancement and systematic validation of an automated pain recognition system. 2013 IEEE international conference on cybernetics (CYBCO),

Wang, F., Xiang, X., Liu, C., Tran, T. D., Reiter, A., Hager, G. D., Quon, H., Cheng, J., & Yuille, A. L. (2017). Regularizing face verification nets for pain intensity regression. 2017 IEEE International Conference on Image Processing (ICIP),

Wang, H., Ding, C. H., & Huang, H. (2010). Multi-Label Classification: Inconsistency and Class Balanced K-Nearest Neighbor. AAAI,

Wang, J., & Sun, H. (2018). Pain intensity estimation using deep spatiotemporal and handcrafted features. *IEICE Transactions on Information and Systems, 101*(6), 1572-1580.

Werner, P., Lopez-Martinez, D., Walter, S., Al-Hamadi, A., Gruss, S., & Picard, R. (2019). Automatic Recognition Methods Supporting Pain Assessment: A Survey. *IEEE Transactions on Affective Computing*.

Wolpert, D. H. (1992). Stacked generalization. *Neural networks, 5*(2), 241-259.

Xie, J., Xu, B., & Chuang, Z. (2013). Horizontal and vertical ensemble with deep representation for classification. ICML 2013, Atlanta.

Xu, M., Cheng, W., Zhao, Q., Ma, L., & Xu, F. (2015). Facial expression recognition based on transfer learning from deep convolutional networks. 11th IEEE International Conference on Natural Computation (ICNC), Zhangjiajie China.

Xu, M., Su, H., Li, Y., Li, X., Liao, J., Niu, J., Lv, P., & Zhou, B. (2019). Stylized aesthetic QR code. *IEEE transactions on multimedia, 21*(8), 1960-1970.

Yang, Q., Zhang, Y., Dai, W., & Pan, S. J. (2020). *Transfer learning*. Cambridge University Press.

Yang, R., Tong, S., Bordallo, M., Boutellaa, E., Peng, J., Feng, X., & Hadid, A. (2016). On pain assessment from facial videos using spatio-temporal local descriptors. 2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA),

Zarit, B. D., Super, B. J., & Quek, F. K. (1999). Comparison of five color models in skin pixel classification. Proceedings International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems. In Conjunction with ICCV'99 (Cat. No. PR00378),

Zhang, C., Wang, P., Chen, K., & Kämäräinen, J.-K. (2017). Identity-aware convolutional neural networks for facial expression recognition. *Journal of Systems engineering and Electronics, 28*(4), 784-792.

Zhang, D., Han, J., Zhao, L., & Meng, D. (2019). Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework. *International Journal of Computer Vision, 127*(4), 363-380.

Zhang, X., Yin, L., Cohn, J. F., Canavan, S., Reale, M., Horowitz, A., Liu, P., & Girard, J. M. (2014). Bp4d-spontaneous: a high-resolution spontaneous 3d dynamic facial expression database. *Image and vision computing, 32*(10), 692-706.

Zhang, Z., Lyons, M., Schuster, M., & Akamatsu, S. (1998). Comparison between geometry-based and gabor-wavelets-based facial expression recognition using multi-layer perceptron. Proceedings Third IEEE International Conference on Automatic face and gesture recognition,

Zhao, R., Gan, Q., Wang, S., & Ji, Q. (2016). Facial expression intensity estimation using ordinal information. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,

Zhou, J., Hong, X., Su, F., & Zhao, G. (2016). Recurrent convolutional neural network regression for continuous pain intensity estimation in video. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, Hawaii, USA.

APPENDICES I

Published Publications Included in this Thesis

No	Papers	Type	Applied in
1	Bargshady, G., Zhou, X., Deo, R. C., Soar, J., Whittaker, F., & Wang, H. (2020). Enhanced deep learning algorithm development to detect pain intensity from facial expression images. *Expert Systems with Applications*, *149*, 113305.	Journal (Q1)	Chapters: 4, 5, 6.
2	Bargshady, G., Zhou, X., Deo, R. C., Soar, J., Whittaker, F., & Wang, H. (2020). Ensemble neural network approach detecting pain intensity from facial expressions. *Artificial Intelligence in Medicine*, *109*, 101954.	Journal (Q1)	Chapter 5.
3	Bargshady, G., Zhou, X., Deo, R. C., Soar, J., Whittaker, F., & Wang, H. (2020). The modeling of human facial pain intensity based on Temporal Convolutional Networks trained with video frames in HSV color space. *Applied Soft Computing*.	Journal (Q1)	Chapter 6.
4	Bargshady, G., Soar, J., Zhou, X., Deo, R. C., Whittaker, F., & Wang, H. (2019). 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)* (pp. 52-56). IEEE Press.	Conference	Chapter 4.
5	Soar, J., Bargshady, G., Zhou, X., & Whittaker, F. (2018, July). Deep learning model for detection of pain intensity from facial expression. In *International Conference on Smart Homes and Health Telematics* (pp. 249-254). Springer, Cham.	Conference	Chapter 2.

Other Publications During Candidature

No	Paper	Paper type
1	Abdar, M., Nasarian, E., Zhou, X., Bargshady, G., Wijayaningrum, V. N., & Hussain, S. (2019, February). Performance improvement of decision trees for diagnosis of coronary artery disease using multi filtering approach. In *Proceedings of the 4th IEEE International Conference on Computer and Communication Systems (ICCS 2019)* (pp. 26-30). IEEE Press.	Conference
2	Zhou, X., Tao, X., Bargshady, G., Gururajan, R., Abdar, M., Chan, K. (2019). A Case Study of Predicting Banking Customers Behaviour by Using Data Mining, *the 6th international conference on behavioral, Economic, and Socio, Cultural Computing*.	Conference
3	Zhou, X., Gururajan, R., Li, Y., Venkataraman, R., Tao, X., Bargshady, G., ... & Kondalsamy-Chennakesavan, S. (2020, August). A survey on text classification and its applications. In *Web Intelligence* (No. Preprint, pp. 1-12). IOS Press.	Journal

APPENDICES II

Published Papers

Expert Systems With Applications 149 (2020) 113305

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Enhanced deep learning algorithm development to detect pain intensity from facial expression images

Ghazal Bargshady [a,*], Xujuan Zhou [a], Ravinesh C. Deo [b], Jeffrey Soar [a], Frank Whittaker [c], Hua Wang [d]

[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] Nexus e-Care, Adelaide, Australia
[d] Victoria University, Melbourne, Australia

ARTICLE INFO

Article history:
Received 30 July 2019
Revised 29 January 2020
Accepted 10 February 2020
Available online 16 February 2020

Keywords:
Facial expression
Pain detection
Deep neural networks
Expert systems in healthcare
Machine learning

ABSTRACT

Automated detection of pain intensity from facial expressions, especially from face images that show a patient's health, remains a significant challenge in the medical diagnostics and health informatics area. Expert systems that prudently analyse facial expression images, utilising an automated machine learning algorithm, can be a promising approach for pain intensity analysis in health domain. Deep neural networks and emerging machine learning techniques have made significant progress in both the feature identification, mapping and the modelling of pain intensity from facial images, with great potential to aid health practitioners in the diagnosis of certain medical conditions. Consequently, there has been significant research within the pain recognition and management area that aim to adopt facial expression datasets into deep learning algorithms to detect the pain intensity in binary classes, and also to identify pain and non-pain faces. However, the volume of research in identifying pain intensity levels in multi-classes remains rather limited. This paper reports on a new enhanced deep neural network framework designed for the effective detection of pain intensity, in four-level thresholds using a facial expression image. To explore the robustness of the proposed algorithms, the UNBC-McMaster Shoulder Pain Archive Database, comprised of human facial images, was first balanced, then used for the training and testing of the classification model, coupled with the fine-tuned VGG-Face pre-trainer as a feature extraction tool. To reduce the dimensionality of the classification model input data and extract most relevant features, Principal Component Analysis was applied, improving its computational efficiency. The pre-screened features, used as model inputs, are then transferred to produce a new enhanced joint hybrid CNN-BiLSTM (EJH-CNN-BiLSTM) deep learning algorithm comprised of convolutional neural networks, that were then linked to the joint bidirectional LSTM, for multi-classification of pain. The resulting EJH-CNN-BiLSTM classification model, tested to estimate four different levels of pain, revealed a good degree of accuracy in terms of different performance evaluation techniques. The results indicated that the enhanced EJH-CNN-BiLSTM classification algorithm was explored as a potential tool for the detection of pain intensity in multi-classes from facial expression images, and therefore, can be adopted as an artificial intelligence tool in the medical diagnostics for automatic pain detection and subsequent pain management of patients.

1. Introduction

Pain is a complex and a rather an individual experience that poses several challenges in terms of its precise measurements and medical diagnosis (Ashburn & Staats, 1999). However, in current methods, when providing evidence-based treatments for pain management, many clinicians utilize valid and standard pain outcome measures. Currently, there exists no effective and reliable method for objectively quantifying an individual's experience of the pain. Clinicians and health care organizations mainly use a patient's self-report to determine the intensity of pain; however, these approaches may not be effective and could be relatively inaccurate. They may have some degree of limitation, and cannot be used with young children, individuals with certain types of

* Corresponding author at: University of Southern Queensland (USQ) Springfield Campus, 37 Sinnathamby Blvd, Springfield Central, QLD 4300, Australia.
E-mail addresses: ghazal.bargshady@usq.edu.au (G. Bargshady), xujuan.zhou@usq.edu.au (X. Zhou), ravinesh.deo@usq.edu.au (R.C. Deo), jeffrey.soar@usq.edu.au (J. Soar), frank@nexusonline.com.au (F. Whittaker), HuaWang@vu.edu.au (H. Wang).

https://doi.org/10.1016/j.eswa.2020.113305
0957-4174/

Applied Soft Computing Journal 97 (2020) 106805

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

The modeling of human facial pain intensity based on Temporal Convolutional Networks trained with video frames in HSV color space

Ghazal Bargshady [a,*], Xujuan Zhou [a], Ravinesh C. Deo [b], Jeffrey Soar [a], Frank Whittaker [a], Hua Wang [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

ARTICLE INFO

Article history:
Received 2 February 2020
Received in revised form 21 September 2020
Accepted 13 October 2020
Available online 16 October 2020

Keywords:
Temporal convolutional network
Facial expression
Pain detection
HSV color space
Video analysis

ABSTRACT

An accurate detection and management of pain, measured through its relative intensity, plays an important role in the treatment of disease and reducing a patient's discomfort. As it is relatively difficult to assess, describe, evaluate and manage the pain level using a patient's self-report, automated pain-detecting tools can provide useful information to assist in the management of pain intensity. This study proposes a new predictive modeling framework that employs a modified Temporal Convolutional Network (TCN) algorithm to recognize the pain intensity prevalent in patients' video frames collected as part of UNBC-McMaster Shoulder Pain Archive and MintPAIN databases. The inputs of the proposed TCN network is composed of the extracted and reduced face image features from a fine-tuned VGG-Face and principal component analysis (PCA) with Hue, Saturation, Value (HSV) color spaces video images. The results of TCN based predictive model, employing a long short-term memory (LSTM) model as well as other state-of-the art models, show that the proposed approach performs faster with a high level of efficiency. This is demonstrated by the low magnitude of error metrics (i.e., Mean Squared Error = 0.0629, Mean Absolute Error = 0.1021, correctness validation results represented by Area under Curve = 85% and accuracy metric = 92.44%). Considering the efficiency of the proposed TCN framework, integrating fine-tuned VGG-Face and PCA with Hue, Saturation, Value (HSV) color spaces video images for pain intensity estimation, the present study affirms that the new method can be adopted as an automatic health informatics tool, mainly for pain detection, and subsequently, implemented in the pain management area.

1. Introduction

An automatic and intelligent face detection and feature recognition technologies have evolved significantly, following the rapid development of computer vision and pattern recognition algorithms, thus, playing a crucial role in designing a real-time system that accurately recognizes human behavior, to support health monitoring and treatment devices. The present study focuses on automatic pain detection through human facial expressions. Pain, often presented as an indicator of physical or mental health issues, is an unpleasant sensation caused by illness, injury or perturbed emotional experience [1]. Managing pain and detecting its relative intensity plays an important role in treatment of disease. Generally, it is difficult to assess or manage pain from a patient's self-reporting documents, as there are some challenges and limitations in the description of the pain level [2]. For example, self-reporting measurements of pain is of little value for infants, individuals with certain types of neurological impairments and dementia, including patients in postoperative care, and patients in intensive care units (ICU).

The human face is a rich source of authentic information, representing a person's social interactions, including the expression of their emotions and their pain [3]. Facial activities are incorporated as a major component of most of the multidimensional behavioral checklists and rating scales used for assessing pain. The Facial Action Coding System (FACS), developed by Ekman and Friesen [4], provides an objective meaning for measuring facial muscle contractions in facial expression. The Prkachin and Solomon Pain Intensity Scale (PSPI) is currently the only metric that can define pain on a frame-by-frame basis with the assistance of FACS [5]. Machine learning algorithms, implemented as

* Correspondence to: 37 Sinnathamby Blvd, Springfield Central QLD 4300, University of Southern Queensland (USQ), Springfield campus, Australia.
E-mail addresses: ghazal.bargshady@usq.edu.au (G. Bargshady), xujuan.zhou@usq.edu.au (X. Zhou), ravinesh.deo@usq.edu.au (R.C. Deo), jeffrey.soar@usq.edu.au (J. Soar), Frank.Whittaker@usq.edu.au (F. Whittaker), HuaWang@vu.edu.au (H. Wang).

https://doi.org/10.1016/j.asoc.2020.106805
1568-4946/

Artificial Intelligence In Medicine 109 (2020) 101954

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

journal homepage: www.elsevier.com/locate/artmed

Ensemble neural network approach detecting pain intensity from facial expressions

Ghazal Bargshady [a,*], Xujuan Zhou [a], Ravinesh C. Deo [b], Jeffrey Soar [a], Frank Whittaker [a], Hua Wang [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

ARTICLE INFO

Keywords:
Ensemble neural network
Pain detection
Facial expression
Deep learning

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.

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), 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

* Corresponding author at: 37 Sinnathamby Blvd, Springfield Central QLD 4300, University of Southern Queensland (USQ) Springfield campus, Australia.
E-mail addresses: ghazal.bargshady@usq.edu.au (G. Bargshady), xujuan.zhou@usq.edu.au (X. Zhou), ravinesh.deo@usq.edu.au (R.C. Deo), jeffrey.soar@usq.edu.au (J. Soar), Frank.Whittaker@usq.edu.au (F. Whittaker), Hua.Wang@vu.edu.au (H. Wang).

https://doi.org/10.1016/j.artmed.2020.101954
Received 21 January 2020; Received in revised form 23 August 2020; Accepted 25 August 2020
Available online 7 September 2020
0933-3657/

A Joint Deep Neural Network model for Pain Recognition from Face

Ghazal Bargshady
School of Management and Enterprise
University of Southern Queensland
Springfield, Australia
Ghazal.bargshady@usq.edu.au

Jeffrey Soar
School of Management and Enterprise
University of Southern Queensland
Springfield, Australia
Jeffrey.soar@usq.edu.au

Xujuan Zhou
School of Management and Enterprise
University of Southern Queensland
Springfield, Australia
Xujuan.Zhou@usq.edu.au

Ravinesh C Deo
School of Agricultural, Computational and Environmental Sciences
University of Southern Queensland
Springfield, Australia
ravinesh.deo@usq.edu.au

Frank Whittaker
Nexus eCare
Adelaide, Australia
frank.whittaker@adelaide.edu.au

Hua Wang
University of Victoria
Melbourne, Australia
Hua.Wang@vu.edu.au

***Abstract*— Pain is a primary symptom of diseases and an indicator of a patients' health status. Effective management of pain is important for patient treatment and well-being. There are some traditional self-reported methods for pain assessment, and automatic pain detection systems using facial expressions are developing rapidly; these offer the potential for more efficient, convenient and cost-effective pain management. In this paper, a joint deep neural network model is proposed to classify pain intensity in four categories from facial images. This study used two different Recurrent Neural Networks (RNN), which were pre-trained with Visual Geometric Group Face Convolutional Neural Network (VGGFace CNN) and then joined together as a network to estimate pain intensity levels. The UNBC-McMaster Shoulder Pain database was used to train and test the proposed algorithm. As a contribution to knowledge, this paper provides new information regarding the performance of a hybrid, joint deep learning algorithm for pain multi-classification in facial images.**

Keywords-facial expressions; pain recognition; deep convolutional network, transfer learning, computer vision

I. INTRODUCTION

Many people who are suffering from chronic pain face periods of acute pain and resulting problems during their illness; adequate reporting of symptoms is necessary for optimal treatment [1]. Pain is measured by self-reports; however, some patients have difficulties in adequately alerting caregivers to their pain or describing the intensity which can then impact adversely on effective treatment. This is because self-reporting of pain is affected by the cognitive and linguistic abilities of the patient[2]; consequently the concept of automatic pain level valuation has been formulated.

Pain and its intensity can be noticed in one's face. Movements in the facial muscles can depict one's current emotional state. Facial expression can provide information about pain and emotional severity[3]. This information can be measured by the Facial Action Unit (FACS) coding [4]. Prkachin and Solomon Pain Intensity (PSPI) metric developed later was based on FACS to measure pain level[4]. Machine learning algorithms and computer vision-based techniques can use FACS and PSPI codes to predict pain and its intensity from patients' faces[4].

Detecting pain level from facial expressions is a challenging task. Luci et al. (2011) provided the UNBC-McMaster Shoulder Pain Expression Archive database that contains 200 video sequences taken from 25 patients who were suffering shoulder pain. The frames were labeled using by the PSPI codes[5]. This database has unbalanced data, and therefore, its analysis and modeling can be very challenging[6]. Unbalanced data refers to classification problems where there are unequal instances for each class [6]. Fig. 1 shows the image frames of a patient in different pain levels based on the PSPI code. As is indicated, recognition of pain levels is not easy.

Recently, research studies used different techniques to predict pain by using the UNBC-McMaster Shoulder Pain Expression Archive database; however, the latest research is focused more on detecting pain in binary classification so identifying pain intensity level in multi-classification is still a challenging issue in the use of this database.

Deep learning architectures such as Convolutional Neural Networks (CNN) or Deep Belief Networks (DBN) have been widely used in face expression analysis, such as emotion recognition, age, and pain estimation [6-9].

In this research, a new joint deep neural network framework which is based on two different hybrids deep networks (CNN + RNN) is proposed. These two networks are trained based on transfer learning to recognize the pain level in four levels, and finally, their outputs are combined [10].

II. RELATED WORKS

There is existing research into automated recognition of effect from facial expressions but until recently, only a few studies have focused on automated pain estimation. Susskind et al. (2008) trained deep belief networks without supervision for recognizing facial action units. That study showed features extracted by learned belief nets was able to

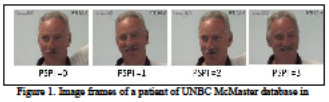

Figure 1. Image frames of a patient of UNBC McMaster database in

Deep learning Model for Detection of Pain Intensity from Facial Expression

Jeffrey Soar[1], Ghazal Bargshady[1], Xujuan Zhou[1], Frank Whittaker[2], and Hua Wang[3]

[1] University of Southern Queensland, Queensland, Australia
[2] Nexus eCare,Melbourne, Australia
[3] University of Victoria, Melbourne, Australia

Abstract. Many people who are suffering from a chronic pain face periods of acute pain and resulting problems during their illness and adequate reporting of symptoms is necessary for treatment. Some patients have difficulties in adequately alerting caregivers to their pain or describing the intensity which can impact on effective treatment. Pain and its intensity can be noticeable in ones face. Movements in facial muscles can depict ones current emotional state. Machine learning algorithms can detect pain intensity from facial expressions. The algorithm can extract and classify facial expression of pain among patients. In this paper, we propose a new deep learning model for detection of pain intensity from facial expressions. This automatic pain detection system may help clinicians to detect pain and its intensity in patients and by doing this healthcare organizations may have access to more complete and more regular information of patients regarding their pain.

1 Introduction

Painful conditions are one of the most common reasons that patients seek health care [1]; some patients may have difficulty providing accurate recall of pain or may be unwilling to disclose their pain. Assessment can be invasive and inconvenient [2].

The needs of the large number of people in chronic pain overwhelms healthcare systems, and continues to worsen [3]. Healthcare providers may be less able to help patients who have chronic pain if there is not a complete understanding of symptoms and some patients such as the very young, or those with particular disabilities or conditions, may not be able to articulate their experiences. Automatic pain management systems have been demonstrated to better help clinicians to detect the level of pain of patients [2].

Faces have evolved to convey rich information for social interaction, including the expression of emotions and pain [4]. Researchers have applied machine learning to the task of automatic pain detection in a real-world clinical setting involving patients suffering pain [5].

In this study, we proposed a new deep learning model for detection of pain intensity from facial expressions. This automatic pain detection model can be