Software Development of Deep Neural Networks Methods Applied to Face-based Emotion Recognition and Pain Detection

Diogo Henrique Vieira Andrade

DISSERTAÇÃO

Supervisor
Ana Filipa Sequeira
Researcher
INESCTEC

Co-Supervisor
Pedro David Carneiro Neto
Researcher
INESCTEC

Software Development of Deep Neural Networks Methods Applied to Face-based Emotion Recognition and Pain Detection

Diogo Henrique Vieira Andrade

Mestrado em Engenharia Biomédica

Approved in oral examination by the committee:

President: Prof. Doutor José Alberto Peixoto Machado da Silva
Referee: Doutor João Ribeiro Pinto
Referee: Prof.ª Doutora Ana Filipa Pinheiro Sequeira

October 31, 2025

Abstract

Emotions are a quintessential part of the human experience. Although not all humans experience or express them equally, emotions often guide one's life. When verbal expression fails, body language and facial expressions can help us convey what we feel. These are generally understood regardless of origin or upbringing, being mostly unconscious reactions and thus less controllable. This has drawn interest from the scientific and software development communities. If our facial expressions are understood almost intrinsically, what are the patterns enabling this comprehension? And more importantly, can computational systems replicate it? This is how Facial Emotion Recognition (FER), and later Automatic Facial Emotion Recognition (AFER) methods began. While they traditionally focus on simpler emotions, happiness, anger, disgust and fear, another, more niche state, has captured recent interest: pain.

Pain is essential for interacting with the environment and understanding if we are under duress. While it may overlap with conventional emotion definitions, there are key factors that differentiate it. It is also crucial for health and well-being management. "How much", "when" and "where" are vital for accurate clinical assessment. Therefore, AFER systems specialized in pain recognition have started to be explored. Although no clinically applicable options exist yet, advancements in AFER, informatics, and artificial intelligence (AI) have accelerated progress in pain recognition.

This work examines the history of this field, and experimentally evaluates models for binary pain recognition. The Biovid and MIntPAIN databases served as the basis for this study, providing facial images and videos captured under pain-inducing stimuli. Images emphasizing the contrast between neutral and painful states were extracted, converted to greyscale, stored, and resized according to model requirements.

Two model families were evaluated: a Resnet18 convolutional model modified for smaller datasets, later expanded to support parallel processing of an input vector of 17 Facial Action Units (FAUs) intensity features as a second approach; and a Data-efficient Image Transformer (DeiT), evaluated as an attention-based alternative.

Model performance was evaluated under multiple conditions, including data augmentations (horizontal flip and random occlusion) varying weight freezing percentages, and different initialization methods. For the Resnet18 models, random initialization and transfer learning from the FER2013 database; for the DeiT, only initialization in the Imagenet dataset was explored. Both sources are widely used in AFER.

The results obtained were significantly below what is found in the literature, with accuracies ranging from 46.3% to 57.1%. The findings highlight the difficulty of simpler models in adapting to stronger modifications of the base images, while the DeiT showed better robustness within similar circumstances. The integration of FAU features consistently helped the performance of the weaker model, even if only slightly. Factors such as limited dataset size and restricted data preparation methods, leading to overfitting, likely contributed to the reduced accuracy. As such, suggested future work should broaden the spectrum of accepted features, to include RGB images, temporal features, and potentially other biosignals to better the classification results.

Resumo

As emoções são essenciais para a experiência humana. Embora nem todos as experienciem ou expressem da mesma forma, as emoções guiam a vida de cada indivíduo. Quando a expressão verbal falha, a linguagem corporal e as expressões faciais permitem comunicar o que se sente. Estas são geralmente compreendidas independentemente de origem ou educação, sendo reações inconscientes e por isso menos controláveis. Isto tem despertado interesse nas comunidades científica e de de desenvolvimento de software. Se as nossas expressões faciais são compreendidas intrínsecamente, quais os padrões que permitem essa compreensão? E, mais importante, podem sistemas computacionais replicá-la? Assim surgiram os métodos de Reconhecimento Facial de Emoções (FER), e mais tarde Reconhecimento Facial Automático de Emoções (AFER). Embora geralmente se foquem em emoções básicas, como felicidade, raiva, nojo, e medo, outro estado tem captado interesse: a dor.

A dor é essencial para as nossas interações com o ambiente e para perceber se estamos em sofrimento. Embora possa coincidir com definições convencionais de emoção, existem fatores-chave que a diferenciam. É também crucial para a gestão da saúde e bem-estar. "Quanto", "quando" e "onde" são informações vitais para uma avaliação clínica precisa. Por isso, têm sido explorados sistemas AFER especializados no reconhecimento da dor. Embora ainda não existam opções clinicamente aplicáveis, avanços em AFER, informática e inteligência artificial (IA) têm acelerado o progresso nesta área.

Este trabalho examina a história deste campo, e avalia modelos para reconhecimento binário da dor. As bases de dados Biovid e MIntPAIN serviram de base para o estudo, fornecendo imagens faciais e vídeos capturados sob estímulos de dor. Imagens que enfatizam o contraste entre estados neutros e dolorosos foram extraídas, convertidas para tons de cinza, armazenadas e redimensionadas de acordo com os requisitos dos modelos.

Foram avaliadas duas famílias de modelos: uma Resnet18 convolucional, modificada para conjuntos pequenos de dados, e posteriormente expandida para incluir processamento paralelo de um vetor com 17 intensidades de Unidades de Ação Facial (FAUs) como segunda abordagem; e um Transformer Eficiente de Imagem (DeiT), avaliado como alternativa baseada em atenção.

O desempenho dos modelos foi avaliado em múltiplas condições, incluindo aumento de dados (espelhamento horizontal e oclusão aleatória), diferentes percentagens de congelamento de pesos e métodos de inicialização distintos. Para os modelos Resnet18, explorou-se a inicialização aleatória e a transferência de conhecimento a partir da base FER2013; para o DeiT, apenas a inicialização na base Imagenet foi utilizada. Ambas as bases são amplamente utilizadas para AFER.

Os resultados obtidos ficaram significativamente abaixo do reportado na literatura, com precisões entre 46,3% e 57,1%. Estes resultados destacam a dificuldade dos modelos mais simples em adaptar-se a alterações mais complexas das imagens, enquanto que o DeiT demonstrou maior robustez em condições semelhantes. A integração das FAUs ajudou consistentemente o desempenho do modelo mais fraco, ainda que de forma ligeira. Fatores como o tamanho limitado dos conjuntos de dados e métodos restritos de preparação, conduzindo a overfitting, contribuíram para a

redução da precisão. Trabalhos futuros deverão alargar o espectro de características consideradas, incluindo imagens RGB, características temporais e potencialmente outros biosinais, de forma a melhorar os resultados de classificação.

Agradecimentos

No final de um ciclo, é sempre difícil saber o que dizer. A quem agradecer, por que ordem o fazer. A verdade é que tudo e todos foram motivações e razões pelas quais cheguei onde e como cheguei. Mas há sempre quem mereça ser salientado.

Em primeiro lugar, e como não podia deixar de faltar, deixo um grande obrigado à Ana Filipa Sequeira e ao Pedro Neto por toda a disponibilidade e apoio. Agradeço pela abertura em todas as reuniões, pela compreensão com as minhas dúvidas e incertezas, e em especial pela preocupação a mim dedicada neste tempo. Também uma palavra de agradecimento ao Rafael Carvalho Maia, que mesmo não sendo parte oficial da equipa de orientação, conseguiu ajudar-me quando não conseguia avançar.

Em seguida, agradecer a quem lidou comigo durante estes 5 anos. Uma fase cheia de dualidades, de choro e riso, de delírio e desespero, de tantas emoções fortes que às vezes parecia impossível lidar com elas. Mas com vocês, tudo se tornou um pouco mais tolerável.

Às minhas meninas de Aveiro, que passaram a licenciatura a lidar com os meus stresses e preocupações, que fizeram sair da ilha um terror menor, e criaram para mim uma segunda casa. Começar a faculdade em plena pandemia, numa altura em que tudo era incerto, não foi o ideal, mas fá-lo-ia de novo se significasse ter o nosso grupo junto. Todas as noitadas, todos os jantares, todos os dias que passámos juntos fizeram perceber que foi a escolha certa.

A quem se juntou no Porto, e apoiou na segunda mudança de vida, fica também o meu obrigado. Foram mais dois anos, mas dois anos bem vividos. Nunca esquecerei a "Veneza Portuguesa", mas fizeram valer a pena a mudança, e tornaram o Porto uma cidade acolhedora e cheia de boas memórias.

Aos que vieram da ilha e decidiram espalhar por Portugal e arredores, um agradecimento e um pedido de desculpas. Nunca fui o melhor a manter relações de longa distância, mas por vocês valeu sempre a pena. E espero que aceitarem os meus desvaneios e esquecimentos de resposta signifique que, para vocês, também vale.

Por fim, agradeço à minha família. Mesmo sem perceberem bem o que estava a fazer ou estudar, sempre me apoiaram. Mesmo quando eu próprio não sabia para onde ia, sempre me ampararam, e sempre lá estiveram, mesmo quando não senti que o merecia. Em especial, obrigado à minha irmã. Espero conseguir fazer-te orgulhosa, agora e para o resto da vida.

Acabadas as pessoas, agradeço também às instituições. À Universidade de Aveiro, que proporcionou a primeira casa fora de casa, e à FEUP, que me deu a segunda.

Finalmente, a todos os que não mencionei em específico, mas que me fizeram vir memórias à cabeça, sorrisos aos lábios e lágrimas aos olhos enquanto escrevia. Estarão sempre presentes na pessoa levou isto até ao fim.

Obrigado a todos!

Diogo Andrade

Contents

1 Introduction **1**
- 1.1 Context . . . 1
- 1.2 Motivation . . . 2
- 1.3 Objectives . . . 3
- 1.4 Structure . . . 3

2 Theoretical Concepts **4**
- 2.1 Pain . . . 4
  - 2.1.1 Definition and Characteristics . . . 4
  - 2.1.2 Types of pain . . . 4
- 2.2 Emotion Recognition . . . 6
  - 2.2.1 Facial Action Units . . . 6
- 2.3 Computer Vision and Deep Learning Methods . . . 8
  - 2.3.1 Convolutional Neural Networks . . . 12
  - 2.3.2 Transformers . . . 14
  - 2.3.3 Training Methods . . . 15

3 State-Of-The-Art **18**
- 3.1 Databases . . . 18
  - 3.1.1 Used Databases . . . 18
  - 3.1.2 Not Used Databases . . . 20
  - 3.1.3 Overall Landscape and Comparison . . . 21
- 3.2 Facial Pain Recognition . . . 22
  - 3.2.1 Face Detection and Preprocessing . . . 22
  - 3.2.2 Feature Extraction . . . 25
  - 3.2.3 Pain Assessment . . . 27
- 3.3 State-of-the-Art Models for Pain Recognition . . . 29

4 Methodologies **33**
- 4.1 Evaluation Framework . . . 33
- 4.2 Dataset Preparation . . . 34
  - 4.2.1 Partitioning . . . 34
  - 4.2.2 Frame Extraction . . . 35
  - 4.2.3 Facial Detection . . . 35
- 4.3 Data Augmentation Techniques . . . 36
- 4.4 Models . . . 36
  - 4.4.1 Altered ResNet-18 . . . 36
  - 4.4.2 ResNet-18 with Action Units (AUs) . . . 37

4.4.3 DeiT-based Pain Detection . . . 38
4.5 Summary . . . 38

5 Experimental Work **40**
5.1 Implementation Details . . . 40
5.2 Dataset Realization . . . 40
5.3 AU Extraction . . . 42
5.4 Implementation of Data Augmentation . . . 42
5.5 Training Procedures . . . 43
5.6 Experimental Protocols . . . 44

6 Results and Discussion **45**
6.1 Experiments Description . . . 45
6.2 Overview of Experimental Results . . . 46
6.3 ResNet18 . . . 46
6.4 ResNet18 + MLP . . . 50
6.5 DeiT . . . 53
6.6 Comparative Analysis . . . 56
6.6.1 Dataset-specific differences . . . 56
6.6.2 Architecture-specific differences . . . 57
6.6.3 Effect of augmentations and freeze levels . . . 58
6.6.4 Training dynamics and overfitting . . . 58
6.6.5 Graphical evaluation of classifier behavior . . . 58
6.7 Difficulties and Limitations . . . 59

7 Conclusions and Future Work **61**
7.1 Conclusions . . . 61
7.2 Finishing Remarks . . . 62
7.3 Future Work . . . 62

List of Figures

2.1 Illustration of multiple FAUs . . . 8
2.2 Schematic representation of a Perceptron. . . . 9
2.3 Structure of the SLP (a) and of the MLP (b). . . . 12
2.4 Schematic representation of a convolution operation . . . 13
2.5 Original ViT model schematic. . . . 14

3.1 Example images and information of the BioVid Database. . . . 19
3.2 Example images of the MIntPAIN Database. . . . 20
3.3 Example images of the UNBC-McMasters Database. . . . 21
3.4 Geometry features (left) vs. Appearance features (right). . . . 26
3.5 Example NRS. . . . 27
3.6 Example VRS. . . . 28
3.7 Example VAS . . . 28

4.1 Example of augmentation application. . . . 36
4.2 ResNet 18 inspired model. . . . 37
4.3 ResNet 18 with parallel branch for AUs. . . . 37
4.4 Data-efficient Image Transformer. . . . 38

5.1 Examples of data augmentation: (Top) Horizontal/Vertical shift; (Bottom) Flip, Occlusion, and Flip + Occlusion. . . . 43

6.1 Accuracy and Loss progression for training and validation sets using the ResNet18 model. . . . 48
6.2 Confusion Matrix for the ResNet18 model (0 - No Pain, 1 - Pain). . . . 49
6.3 ROC Curve for the ResNet18 model. . . . 49
6.4 Genuine vs Impostor score distributions for the ResNet18 model. . . . 50
6.5 Accuracy and Loss progression for training and validation sets using the MLP model. 52
6.6 Confusion Matrix for the MLP model (0 - No Pain, 1 - Pain). . . . 52
6.7 ROC Curve for the MLP model. . . . 53
6.8 Genuine vs Impostor score distributions for the MLP model. . . . 53
6.9 Accuracy and Loss progression for training and validation sets using the DeiT model. 54
6.10 Confusion Matrix for the DeiT model (0 - No Pain, 1 - Pain). . . . 55
6.11 ROC Curve for the DeiT model. . . . 55
6.12 Genuine vs Impostor score distributions for the DeiT model. . . . 56
6.13 Dataset comparison. . . . 57

List of Tables

3.1 Comparison of pain-related databases. “x” indicates the presence of a given modality. (Phasic - short, externally induced pain; Tonic - longer, sustained pain.) . . . 22
3.2 Summary of representative models for binary pain recognition. 32
3.3 Summary of representative models for multiclass pain recognition. 32

5.1 Subject distribution across partitions for the BioVid dataset. 41
5.2 Subject distribution across partitions for the MIntPAIN dataset. 41
5.3 Frame distribution for the BioVid dataset. 42
5.4 Frame distribution for the MIntPAIN dataset. 42

6.1 Test accuracy (Mean ± Standard Deviation) across datasets. 46
6.2 ResNet18 (baseline) accuracies on the BioVid dataset with different augmentations and freeze percentages. 47
6.3 ResNet18 (FER2013+) accuracies on the BioVid dataset with different augmentations and freeze percentages. 47
6.4 ResNet18 (baseline) accuracies on the MIntPAIN dataset with different augmentations and freeze percentages. 47
6.5 ResNet18 (FER2013+) accuracies on the MIntPAIN dataset with different augmentations and freeze percentages. 48
6.6 ResNet18 + MLP(baseline) accuracies on the BioVid dataset with different augmentations and freeze percentages. 50
6.7 ResNet18 + MLP(FER2013+) accuracies on the BioVid dataset with different augmentations and freeze percentages. 51
6.8 ResNet18 + MLP (baseline) accuracies on the MIntPAIN dataset with different augmentations and freeze percentages. 51
6.9 ResNet18 + MLP (FER2013+) accuracies on the MIntPAIN dataset with different augmentations and freeze percentages. 51
6.10 DeiT accuracies on the BioVid dataset with different augmentations and freeze percentages. 54
6.11 DeiT accuracies on the MIntPAIN dataset with different augmentations and freeze percentages. 54

7.1 Review of the best accuracy scores, in percentage (%), per dataset, per model. . . 61

Abreviaturas e Símbolos

AAM	Active Appearance Model
AFER	Automatic Facial Emotion Recognition
AI	Artificial Intelligence
ANN	Artificial Neural Network
APA	American Psychological Association
AU	Action Unit
AUROC_AUC	Area Under the Receiver Operating Characteristic Curve
CNN	Convolutional Neural Network
DL	Deep Learning
ECG	Electrocardiogram
EEG	Electroencephalograms
EMG	Electromyogram
FACS	Facial Action Unit Coding System
FAU	Facial Action Unit
FER	Facial Emotion Recognition
GPU	Graphical Processing Unit
HOG	Histogram of Oriented Gradients
IASP	International Association for the Study of Pain
LSTM	Long Short-Term Memory Network
ML	Machine Learning
MLP	Multi-Layer Perceptron
MTCNN	Multi-Task Cascaded Convolutional Network
NRS	Numerical Rating Scale
PSPI	Prkachin and Solomon Pain Intensity
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
SGD	Stochastic Gradient Descent
SLP	Single-Layer Perceptron
SVM	Support Vector Machine
VAS	Visual Analogue Scale
ViT	Vision Transformer
VRS	Verbal Rating Scale

Chapter 1

Introduction

This chapter will serve as an introduction to the main topics and motivations behind the dissertation, the objectives of the work, and a summary of the structure of the document.

1.1 Context

The concept of emotion has changed over the years. Even in our age, and despite numerous efforts to create a functional and consensual definition, professionals in psychology were still not able to come up with such a concept, instead creating multiple different frameworks that can be used to regularize work around describing and identifying specific emotions.

One of the first influential studies related to the analysis of human expression of emotions was Charles Darwin's "The Expression of the Emotions in Man and Animals", where Darwin argued that several, considered base, emotions had parallel facial expressions between not only different cultures, but also different species [1]. This similarity in emotional display paves the way for basic emotion recognition.

Pain, however, differentiates itself from other regular emotions, such as happiness, sadness, or anger. Although often considered in equal regards to other traditional emotions, in research and clinical terms it is often separated. While it overlaps considerably with the typical notion of an "emotion", both conceptually and functionally, one major thing that separates pain from the remaining emotions is the existence of dedicated nociceptive systems, that help the body detect noxious stimuli. This makes pain unique, as it is the only emotional experience with full neurological systems dedicated to it [2]. These systems were already present millions of years ago and are thought to have contributed to the foundation of the currently defined pain experience.

The level and type of pain reported by or for a patient can dictate the rest of their treatment. Modern methods still rely on self-reporting for this analysis. In scenarios where the patient is not communicative or cannot express in a comprehensive or effective way how they are feeling, due to impairment, language barriers, or other factors, this is no longer a viable option. Here, the report of an outside observer must be relied upon. Usually, health experts are the ones reporting in these cases, after thorough observation, but even then, they are prone to errors coming from subjective

biases and opinions that may affect how they perceive pain, and how they choose to relay that information.

Automatic pain assessment tools, would therefore be an effective way of complementing human caregivers in this regard, serving as an impartial counterpart of the evaluation, improving communication between them and the patient, and ultimately the quality of their service and pain management. Although multiple bio-signals can potentially be used for this purpose, such as electrocardiograms (ECG) [3] and electroencephalograms (EEG) [4], analyzing the facial expression of a subject is already a reliable enough method to estimate the level of pain they could be experiencing, with the accuracy of methods relying solely on this uni modal approach rivaling those that employ more of these factors. Since this is not a novel concept, being generally considered a specialized version of facial emotional recognition (FER), multiple attempts have been made to create an algorithm capable of efficiently and accurately live predicting the degree of pain a person is feeling, these attempts evolving and expanding with the progress made to programming and technology itself.

Unfortunately, as evidenced by the lack of a system in the market that is able to perform this task, no existing algorithm has come close to the reliability expected from clinically applicable products. As such, much progress and research are yet to be made in this domain, before a model capable of being employed on-field is made.

1.2 Motivation

Improving caregiving techniques and resources is an area of continuous research, particularly in enhancing communication between caregivers and patients. Clear communication about pain levels is essential to effective care, yet many patients are unable to communicate their discomfort verbally due to physical or cognitive limitations. In this context, automatically quantifying a patient's pain through clear, observable indicators presents a valuable opportunity for quicker and more adaptive responses to pain, ultimately improving care quality and responsiveness. Additionally, remote monitoring of pain could allow healthcare providers to respond more flexibly to rising pain levels, conserving resources that would otherwise be dedicated to continuous monitoring. Facial expression analysis is emerging as a particularly effective and non-intrusive method for detecting pain, as facial cues often communicate discomfort more accurately than self-report, or even external report, alone.

Although not yet mature enough for clinical application, automatic pain assessment has gained traction from progress in automatic facial expression recognition (AFER), while also requiring specialized research due to its distinct challenges. Notably, automatic pain assessment has yet to address specific gaps: distinguishing between chronic and acute pain expressions, establishing a standardized pain scale, identifying consistent facial indicators of pain, and overcoming limited dataset availability. Pain expressions are more varied and subtle than other emotions, often involving less pronounced facial movements and more individualized responses. Additionally, automatic pain assessment research faces a shortage of datasets specifically dedicated to pain assessment, and

even those that do exist are often difficult to access, relatively small in size, and not standardized in neither annotation, scale nor data modalities acquired.

As artificial intelligence (AI) and deep learning (DL) continue to evolve, their applications in healthcare, particularly in fields such as automatic pain assessment and AFER, are expanding. Yet, there is considerable room for improvement in automatic pain assessment, where robust, efficient models and more comprehensive databases could lead to meaningful breakthroughs. Advancing automatic pain assessment research would improve our understanding of pain expression itself, and establish a standardized pain classification system, paving the way for reliable, adaptable algorithms that could one day transform patient monitoring and clinical pain assessment.

1.3 Objectives

The primary objective of this work is to analyze and explore the current state of pain assessment based on FER, focusing on the methods used and the resources available. Although the initial project plan aimed at exploratory cross-dataset re-standardization, the difficulties of dataset compatibility and resource limitations, namely due to the differences in recording setup and in biosignals captured, made it so that the work was redirected towards analyzing current methodologies and identifying barriers to automatic pain recognition, and ultimately establishing a potential framework for model comparison in pain detection.

1.4 Structure

This document is organized into several chapters that provide comprehensive information necessary for understanding the topic and the research conducted.

Chapter 2 provides a theoretical foundation for the study, introducing key concepts and tracing the evolution of pain assessment. It covers the history and advancements in automatic facial emotion recognition, with a focus on recent developments, and then integrates these fields to form the basis for automatic face-based pain recognition, the central theme of this work.

Chapter 3 reviews the current state of the art in the field, including the most influential databases, with focus on those used in this study. It also examines notable models and approaches, emphasizing deep-learning and video/image based methods.

Chapter 4 outlines the methodologies applied in our independent study and experimentation with the databases, the procedures and the models used. Chapter 5 then resumes the specifics of the work that was done, implementation details and the specifics of putting methodologies in practice.

Chapter 6 presents and discusses the results obtained and key observations from our findings.

Finally, Chapter 7 offers the dissertation's end discussion, final conclusions, and recommendations for potential future research directions in this area.

Chapter 2

Theoretical Concepts

This chapter establishes the theoretical and background knowledge deemed important to fully understrand the rest of the document.

2.1 Pain

This section introduces the definition and modes of perception of pain.

2.1.1 Definition and Characteristics

According to the International Association for the Study of Pain (IASP) [5], the currently accepted definition for "pain" is as follows: "An unpleasant sensory and emotional experience associated with or resembling that associated with actual or potential tissue damage". Although often equated with nociception, a term used to describe neural processing noxious stimuli [6], the two are not synonymous nor mutually exclusive. [2, 7] Pain is a complex process, involving multiple sensory, affective, and cognitive components. For this reason, the characterization and categorization of pain is a difficult but crucial task. [8]

2.1.2 Types of pain

The IASP has subdivided pain into three types: Nociceptive, Neuropathic and Nociplastic.

Nociceptive, perhaps the most familiar for everyday application, refers to pain caused from actual or threatened damage to non-neural tissue, and is marked by the activation of nociceptors. [9]

Neuropathic pain is defined as pain caused by a lesion or disease of the somatosensory nervous system. [9, 10] This and Nociceptive pain were categories initially conceived to fully contrast each other. Nociceptive representing pain perceived with a normally functioning, undamaged somatosensory system, and neuropathic pain emerging from damage or disease within that system. In that, they were believed to account for all forms of pain.

However, in 2017, IASP introduced a new denomination for the cases deemed outside of these criteria, Nociplastic Pain. [9, 11] This term describes pain arising from altered nociception, where no clear sign of actual or threatened tissue damage capable of activating nociceptors, nor evidence of lesion or disease of the somatosensory system, is perceived. Due to its ambiguous mechanisms, nociplastic pain can occur in combination with nociceptive pain, presenting additional complexity in both diagnosis and treatment.

Pain can also be categorized according to various other metrics, some of them including duration, mechanism, location, and cause. For the purposes of automatic pain detection, the distinction parameter that is taken into account, if one is, is the duration. Within this criterion, pain can be divided into three types, from shortest to longest lasting: transient, acute, and chronic. [12]

Transient pain is usually caused by the activation of nociceptive receptors present on various tissues, like the skin, at the location of the stimuli. It is not necessarily accompanied by any damage and serves more as an alarm to the body that a possible threat to tissue integrity is imminent. As such, if the perceived danger is dealt with, the pain will subside. It is the most common type of pain, being mostly a product of day-to-day life activities, and therefore rarely justifies closer inspection, being at most incidental to other aspects or conditions. Examples include paper cuts, touching a hot surface, and bumping a corner. [13]

Acute pain is similar to transient pain, resulting from stimulation of nociceptors at the site of stimulation, except now there is tangible damage to the tissue. Usually, pain persisting for a maximum of 30 days is classified as acute. The body is often able to heal itself without eliciting intervention, and the perception of pain stops even before the body is fully healed. If medical intervention happens, it usually consists of pain prevention or assistance to the natural healing process, potentially shortening it or ensuring its efficiency. Broken bones, sprains, and post-surgical pain are some possible examples of acute pain. [13]

Finally, chronic pain, is regarded as persisting past normal healing time [14], or, more specifically and according to the latest version of the International Classification of Diseases (ICD-11), is classified as pain that persists or recurs for longer 3 months. [14, 15, 16] Pain lasting between 30 days and 3 months is referred to as subacute pain. This transitional phase often reflects slower healing processes, and medical intervention may be needed to prevent progression to chronic pain. Chronic pain itself is commonly triggered due to injuries or diseases, being even potentially divided into benign, meaning non-cancer-related, and malignant, meaning resulting from cancer. The continued duration of the pain is a sign that the healing capabilities of the body are not enough to solve the issue, or that the body is ill-equipped to deal with this specific case. The nervous system is often involved in injuries resulting in this type of pain, being damaged and unable to restore its original state. Chronic pain is unrelenting, leading people to always seek health care evaluations, but it can often not be treated adequately, or even be impossible to treat in such a way as to entirely remove the pain, providing only temporary relief from it. Chronic pain is therefore distinguished by the body's inability to restore homeostasis independently, unlike transient or acute pain, where healing typically occurs without long-term intervention. [13]

In this, being able to recognize when someone is in pain, and in how much of it, is essential to provide the best care possible.

2.2 Emotion Recognition

Understanding the emotional state of the individuals around us is at the crux of every social interaction that any human has. It stands to reason then, that it would be a topic that intrigued and captivated researchers throughout the centuries. As far back as ancient Greece, philosophers such as Aristotle and Hippocrates have attempted to explain human emotion in terms of bodily humors and temperaments, proposing early frameworks for categorizing affective states. Later figures like Charles Darwin [1] and Paul Ekman[17, 18, 19] further developed the study of emotions by linking them to observable physical manifestations, such as facial expressions and body movements. While it is relatively straightforward to observe that emotions are often accompanied by physical manifestations, it is far more complex to determine which specific manifestations correspond to which emotions, or why particular feelings lead to certain expressions. The challenge lies not only in identifying these physical cues but also in understanding the intricate interplay between physiological responses and emotional experiences.

According to the American Psychological Association (APA), an emotion is currently defined as a "complex reaction pattern, involving experiential, behavioural, and physiological elements, by which an individual attempts to deal with a personally significant matter or event". [20] When it pertains to computer vision and human-computer interaction, facial expressions of emotion take center stage as interpretable reactions among these patterns.

Facial expressions are among the most immediate conveyors of emotion, and are one of the most fundamental means of non-verbal communication. Additionally, their ease of manufacture and acquisition make them ideal candidates for computational analysis and daily integration. Proliferation of cameras in laptops, smartphones and other devices elevates the potential uses of facial data for emotional analysis to new heights, making it a major area of research within affective computing. Of course, challenges arise as to how to systematically map this visual information into meaningful categories, and to determine the correct associated responses, whilst maintaining as much transparency and interpretability as possible. To that end, Facial Action Units (FAUs) arise as one of the more widespread and commonly used methods.

2.2.1 Facial Action Units

The Facial Action Unit Coding System (FACS) had its publicized origin in 1978, by the hands of Paul Ekman and Wallace V. Friesen [18], and reviewed in 2002 by Ekman, Friesen and Joseph Hager [19]. Before delving into the system itself, it is important to understand the foundational concept behind it. Namely, that certain emotions give rise to facial expressions that are innate and universal across different cultures.

The modern exploration of idea stems from one of Ekman's more influential works, "Universals and Cultural Differences in Facial Expressions of Emotions", where he challenged previous

studies that denied the existence of such universality. According to Ekman, they failed to consider "display rules", a term coined by him on the same work, to refer to the culture specific norms that would influence how and when emotions are expressed. The existence of these "display rules" was used to explain why, when considering the same scenarios, different cultures may present different expressions. [21]

Through a combination of societal norms and expectations, the same situations may in fact elicit different emotions, typically aligned with the expressions observed. Furthermore, those same norms could lead individuals to conscious or unconsciously take over to alter their facial expressions, resulting in mixed or transitional expressions that do not reflect the initial emotional response.

Ekman thus argued that, while the resulting facial expressions from the initial emotional stimuli were universal, the cultural modulation of those expressions was not. In doing so, he essentially bridged the perspectives of cultural relativists like Birdwhistell [22], and universalists like Darwin [1], though his stance remained fundamentally universalist.

His work has had a profound influence on fields such as psychology and, more relevant to this discussion, computer vision, particularly in standardizing methods for facial expression recognition. Nevertheless, both his early and more recent contributions continue to draw criticism and provoke debate within academic circles. Popularized among the public by the animated film Inside Out, Ekman's theory posits the existence of six basic emotions: "happiness", "sadness", "anger", "fear", "surprise", and "disgust". Later, "contempt" was added to this list, expanding the framework. [17, 21]

Having firmly established his theoretical stance, Paul Ekman continued his research into facial expressions. In 1978, again with Wallace V. Friesen, he published the first iteration of the Facial Action Coding System (FACS). [18] It was devised as a method for describing any facial movement, grounded in observation of the anatomical basis of these movements across various forms of media.

More importantly, FACS provided a common nomenclature that could be used throughout all modalities of facial movement research, allowing for clearer communication between disciplines. As the name suggests, it works off a system of Action Units (AUs), or Facial Action Units (FAUs), fundamental actions of specific facial muscles. Each AU is identified by a number, a short and descriptive label, and the anatomical structures involved. Additionally, AUs are rated with an alphabetical five-point intensity scale, going from A to E, with A being the minimum intensity and E the maximum. Figure 2.1 presents some visual examples of the main AUs.

Figure 2.1: Illustration of multiple FAUs [23]

Although FACS by itself bears no emotion specific connotations in any of its descriptors, it has been used to help with the interpretation of non-verbal signals, particularly facial expressions associated with emotions. By making emotion-based inferences for any individual or combination of AUs, researchers can construct a rough road-map of the components commonly associated with specific emotions. While the system only accounts for visible and obvious movements, discarding non-movement-based alterations, it remains one of the most widely used and trusted frameworks in this field.

The fields of AFER and computer vision benefited greatly from having an easy to use and standardize dictionary of descriptors, that could be generalized to many a case.

2.3 Computer Vision and Deep Learning Methods

Deep Learning (DL) is a specialized kind of Machine Learning, itself a branch of Artificial Intelligence (AI), based on Artificial Neural Networks (ANNs), designed to model complex patterns in data, loosely inspired by and attempting to replicate how the human brain functions. What separates it from typical Machine Learning (ML) is exactly the use of many layers of ANNs, that are capable of capturing features impossible through more conventional means. [24]

These ANNs are composed of several nodes, which are the artificial equivalent of neurons, that are themselves organized in layers, meaning the network can learn hierarchical representations of data. These nodes are interconnected through the equivalent of synapses, here called weighted connections, processing and transmitting information to each other. DL models, notably, have overcome the limited ability of traditional machine-learning techniques to process raw data, being able to automatically extract necessary features from it. [24]

The simplest possible version of an ANN is called a Perceptron [25], an example of which is represented in Figure 2.2. It consists of a single node, that receives input and produces output. More complex ANNs are therefore composed of multiple of these Perceptrons.

To calculate their final output, the Perceptron has in consideration all its inputs (x) and the weights (w), representing how strongly that particular input will impact the decision, associated, along with the inherent bias (b) of the Perceptron, which is a constant. A linear combination of the weighted inputs, added with the bias, is then passed through an activation function (a), which applies a nonlinear transformation, determining the final output value of the Perceptron. This process is translated mathematically in Equation 2.1.

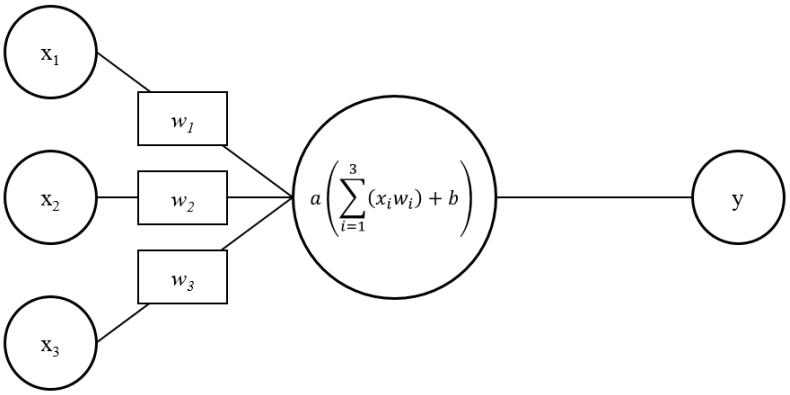


Figure 2.2: Schematic representation of a Perceptron.

$$y = a\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{2.1}$$

There exist multiple options to choose from when considering which activation function to use, and depending on the situation, one may be more appropriate than the rest.

The most elementary one is called the Step Activation Function (Equation 2.2). It outputs either 0 or 1, given an input lower or equal/higher than 0, respectively. Variations can be made and given different outputs and deciding thresholds, but all come back to this basic concept.

$$a(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{2.2}$$

The Linear Activation Function (Equation 2.3) is another extremely simple example, and arguably the most basic one. Here the output is the unaltered version of the input. Because of this, it is also referred to as the Identity Function.

$$a(x) = x \tag{2.3}$$

The Step and Linear Activation Functions are the most rudimentary examples of their kind. The Step Function is not differentiable, and the Linear Function does not permit the network to learn nonlinear relationships. They are therefore mostly employed in much older models, or when exploring the basic functionalities of ANNs.

The Logistic Sigmoid, or simply Sigmoid, Activation Function (Equation 2.4) represents a step up in complexity. It results in an output between 0 and 1, being extremely useful for binary classification problems. It is notable for its polarizing behavior, saturating across most of its domain. It saturates to a higher value when x is very positive, and to a low value when x very negative, and is only really sensitive to the actual input value when x is near 0.

$$a(x) = \frac{1}{1+e^{-x}} \tag{2.4}$$

The Hyperbolic Tangent, or Tanh, Activation Function (Equation 2.5) behaves in a similar fashion to the Logistic Sigmoid, with its domain being instead between -1 and 1. It also has the properties that make the Sigmoid Function perform so well in binary classification, even typically having a better implementation due to performing similarly to the Identity Function when near 0.

$$a(x) = \tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \tag{2.5}$$

These two are closely related, given that they are both sigmoidal functions, and that the Tanh function is, essentially, a scaled and shifted version of the Sigmoid function, as illustrated in Equation 2.6, where σ refers to the Logistic Sigmoid Function and tanh refers to the Hyperbolic Tangent Function. Despite still being used in modern networks, these two functions have fallen out of fashion due to suffering from vanishing gradients. This is when the gradients used to update weights during the models training become extremely small, leading to minimal updating or learning from the weights.

$$\tanh(x) = 2\sigma(2x) - 1 \tag{2.6}$$

The Rectified Linear Unit, or ReLU, Activation Function (Equation 2.7) is a more recent addition to the roster, but an extremely popular one, in part due to not being as susceptible as the previous ones to vanishing gradients, maintaining them for positive inputs. In short, if the value inserted is bigger than 0, it goes through unaltered. But if it is equal or lower than 0, it is outputted as 0, possibly leading to the deactivation of that specific neuron connection. It can be used to make a more efficient and direct ANN.

$$a(x) = \max(0,x) \tag{2.7}$$

The simple and straightforward application of the ReLU function has been extended through multiple variants, each with their own merits and applications.

The Leaky ReLU [26] (Equation 2.8)is considered an improved version of ReLU. The α is commonly set to a small value like 0.01, and so, instead of an absolute zero for negative inputs,

there is a slight positive slope. This variation arose from a problem faced by some architectures using ReLU, where some units were never updated due to being always inactive and essentially becoming dead weight to the model, and decreasing its ability to fit the data. Leaky ReLU then circumvents this by keeping the opportunity for improvement open for negative input values.

$$a(x) = \begin{cases} x & x \geq 0 \\ \alpha x & x < 0 \end{cases} \tag{2.8}$$

The Parametric ReLU [27] (Equation 2.9) is another ReLU variant, or more aptly put, a Leaky ReLU variant. Instead of taking the α value as a constant, it treats it as a parameter to be learned. It is usually applied when the more computationally economic Leaky ReLU still fails to solve the problem of deactivated units.

$$a(x) = \begin{cases} x & x \geq 0 \\ \alpha x & x < 0 \end{cases} \tag{2.9}$$

The Exponential Linear Units [28], or ELUs, Activation Function (Equation 2.10) is also a ReLU variant focused around the negative part of the function, using a logarithmic curve to define them. It can be more desired than its counterparts due to its smoother transition, and the logarithmic nature of the curve may be enough to nudge weights and biases in the right direction.

$$a(x) = \begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases} \tag{2.10}$$

While ReLU and its variants alleviate the vanishing gradient problem, they introduce new and sometimes in fact opposite challenges, such as exploding gradients, which can addressed through careful initialization and normalization techniques.

Finally, there is the Softmax Activation Function (Equation 2.11), which is somewhat unique. It normalizes raw outputs into a probability distribution, meaning the sum for the output related to all classes equals 1. It is then useful for multi-class classification tasks, as it can process the relative probabilities of belonging to each considered class. For this reason it is usually used only within the last layer of the network [29].

$$a(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \tag{2.11}$$

Although they are newer and not exactly staples of the area, it is also relevant to mention the Swish [30] and Mish [31] Activation Functions. While more recent than the other ones listed here, they show undeniable promise, and serve as reminders that newer aspects of this work are being constantly worked upon by the community.

With this in mind, it is important to note that a a single Perceptron, or even a Single-Layer Perceptron (SLP) (Figure 2.3 (a)) does not qualify as Deep Learning, since there is only one layer of weights between the inputs and outputs. The key for operating DL is the stacking of multiple

layers of artificial neurons, connected and with different characteristics, resembling the layered connectivity of biological neurons. A Multi-Layer Perceptron (MLP) (Figure 2.3 (b)), is then the first logical step, introducing at least one hidden layer (meaning a layer that is neither the input nor the output).

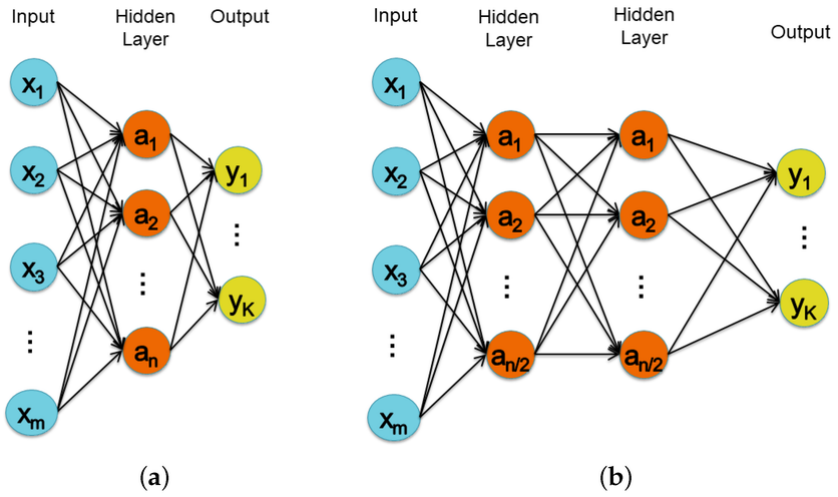


Figure 2.3: Structure of a SLP (a) and a MLP (b).[32]

Among the most common types of networks used for deep learning are Convolutional Neural Networks (CNNs), particularly when the data consists of images. They are remarkable for their capability of reaching excellent performances and extracting hierarchical features and patterns from the raw data.

2.3.1 Convolutional Neural Networks

Despite numerous advancements in the field of deep learning, at the time of writing, CNNs are still considered state-of-the-art for tasks related to signal processing in general, but more specifically, for image classification. At their core, these algorithms contain multiple convolutional layers and pooling layers, often followed by fully connected layers, organized in order to find and single out the most prominent features of the data, combining them to ultimately produce accurate predictions. [8]

Convolutional layers are the lifeblood of CNNs. A convolutional layer applies a set of grid-like filters (kernels), that slide across the input, more often than not an image, and turn it into a feature map. This happens through the mathematical convolution of the input data and the filter.

The kernel, usually with smaller dimensions than the input, is applied to the start of the input, crafting the first component of the feature map. It then moves a number of grid-spaces equal to the stride specified, and the same thing occurs, computing the convolution at this position to produce the second component. This process repeats until the entire input has been covered.

Depending on the requirements of the CNN, padding can be applied to the input in order to control the dimensions of the resulting feature map. It consists of adding extra layers of border

values, usually zeroes, around the input. This helps to develop deeper networks, as without it, the data would shrink after each successive convolution, limiting the depth of CNN to when the input becomes smaller than the kernel, making further convolution impossible.

Figure 2.4 illustrates a schematic representation of a 2×2 kernel applied to a 4×4 input, which with a stride of 1 results in a 3×3 feature map.

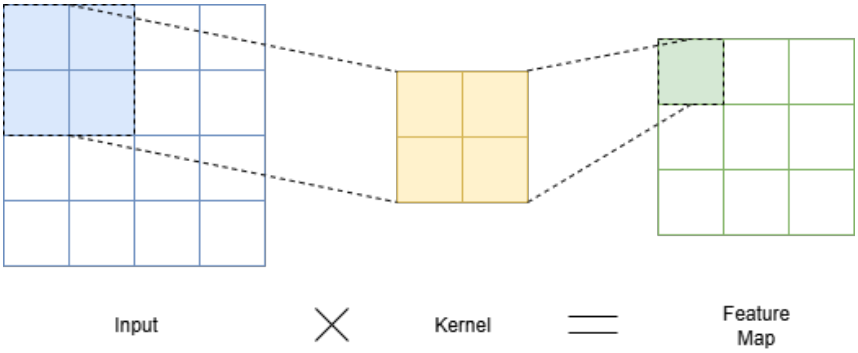


Figure 2.4: Schematic representation of a convolution operation

Pooling layers have the function of condensing regions of a feature map, defined by a window size with specified dimensions, for example 2×2 or 3×3, into just one representative value. This reduces the spatial dimensions of the feature map while keeping only the most important information.

The two most common types of pooling layers are Average Pooling and Max Pooling layers. Average Pooling layers will calculate the mean value of the elements within each window, while Max Pooling layers will simply select the maximum value. These layers are useful for downsampling the data, a practice that leaves it more compact and less computationally expensive to further process.

Fully connected, or dense, layers are usually found near or at the end of the CNN.In these, every node from the previous layer is connected to every node of the current one. They try to integrate the features extracted throughout the network and produce the final output, depending on the original purpose of the CNN. Fully connected layers flatten the feature maps constructed, combining the multiple representations made from the raw data to make a prediction.

Importantly, there are also Activation layers. These are not composed of nodes, but instead apply an activation function, likely one of the those already listed, to the outputs of the preceding layers. Their role is primarily to introduce non linearity to the model, enabling the interpretation of complex relationships within the data. Activation layers usually appear after convolutional layers, in effect "preparing" the feature maps for the next layer, or after the fully connected layer, affecting the final outputs of the model.

Some CNN architectures also incorporate Dropout layers, which randomly deactivate a fraction of nodes during training. This helps prevent overfitting, encouraging the network to learn more robust and redundant feature representations.

Archetypes like VGG-16 [33] and GoogleNet [34] are examples of CNNs used for pain recognition and classification specifically, achieving high accuracies, as expected of their performance with other AFER related work.

2.3.2 Transformers

Another archetype that has been gaining popularity in recent years is the Transformer. While the origins of CNNs can arguably go back to the 1980s, or even further, considering the human brain is their core inspiration, transformers were first introduced in 2017 by Google Brain researcher Ashish Vaswani. [35] It was an architecture envisioned to rival recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), which were at the time the state-of-the-art in sequence modeling and transduction problems such as language modeling and machine translation. Unlike these previous models that relied on convolution and recurrent networks, the Transformer relies on a mechanism called self-attention to model the relationships between input elements, regardless of their distance within the data.

This self-attention, or intra-attention, mechanism allows the model to assign a level of importance to each element of the input, in relation to all other elements. It maps a query vector, as well as a set of key-value paired vectors, to each element. The output is also a vector, made from the weighted sum of the values. The weight, or attention, assigned to each value is computed via a compatibility function of the query and corresponding keys. This way, the model can compute dependencies and contextual relationships efficiently

This new approach even surpassed all previously reported alternatives in certain benchmark tasks, while also demanding less computational cost, effectively establishing itself as state-of-the-art in sequence modeling. Naturally,extending Transformers to domains beyond text became a major avenue of research. In 2020, Alexey Dosovitskiy, another Google Brain researcher, achieved just that. [36] The denominated Vision Transformer (ViT), whose structure is shown in Figure adapts the same attention-based principles of the original, applying them to images instead.

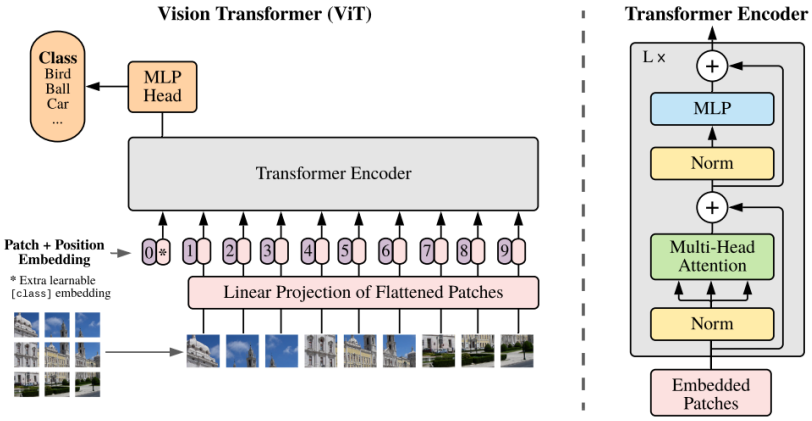


Figure 2.5: Original ViT model schematic.[36]

The concept behind the ViT is to treat the images as a sequence of patches, for example 16×16, analogous to how a sentence is treated as a sequence of words. The input image is divided into non-overlapping patches of the established size. Each patch is flattened and projected into an embedding vector, with each patch preserving its spatial information through additional positional embeddings.

These are then fed to encoder layers, each composed of a multi-head self-attention mechanism, where several self-attention operations, or 'heads', are computed in parallel. This time, they relate each patch to all others in order to learn their dependencies. Each head can focus on different kinds of relationships between patches, and their outputs are combined to enrich the representation. Then, this combined output is passed through a network of fully connected layers with non linear activation functions, and finally normalization layers to stabilize training.

To finalize, a learnable classification token is assigned to the sequence of patch embeddings. The encoder output corresponding to this token is used as the final image representation for classification tasks.

Unlike CNNs, which have built-in inductive biases, ViTs do not rely on convolutions. Instead, all feature interactions are captured through attention. This allows ViTs to achieve competitive or superior performance on image classification benchmarks, just like its originator, provided it is trained on sufficiently large datasets. However, the complexity of self-attention with respect to the number of patches and the absence of inductive biases make large-scale data and careful training strategies important for optimal performance. [36]

2.3.3 Training Methods

After choosing the architecture, the model needs to be trained with data relevant to the task it was designed for. Despite architectural differences, the training processes of most deep learning models are similar in concept and execution, with only finer details, like optimization methods for the models, differing. The most common approach to training a model is via supervised learning. In this process, the model is trained on annotated datasets, containing both the input and known "ground-truth" labels. The programmer must specify the acceptable margin of error, or loss, between the predicted outcomes and the previously known ground-truth. This limits the datasets usable for training to annotated datasets. The quality and type of annotation may hinder the training, or in some cases make it easier. Because of this, choosing the correct training data is essential for this type of training to perform to its fullest potential. [24, 29]

Training occurs by iteratively shifting the weights initially defined for the system, to try and minimize the chosen loss function. To properly calculate the impact a certain shift would cause to the overall error, gradients of the loss function, relating to the parameters, are calculated via backpropagation. The weights will then be altered according to these gradients, opposing the increase of the loss function.

In most cases, the stochastic gradient descent (SGD) procedure is applied. It consists of introducing some examples as input to the system, processing them to obtain the outputs and respective error, and using those results to calculate the gradients that will be used to alter the weights. This

process is repeated for other samples until the loss calculated stabilizes. It's more recent variations, Adam [37] and AdamW [37] are also frequently used, with AdamW in particular being notable for improving generalization. Instead of processing the entire dataset at once, models are trained on smaller batches of data, making computation more efficient.

The training process typically involves splitting data into three subsets, the training set, used for most of the process and the first thing fed to the model, the validation set, used to tune the learning parameter of the model, such as learning rate, batch size and number of epochs, and monitoring for overfitting. The model is then exposed to a different set of data, one that has not been used for training, as a means of final evaluation of the ability of general classification of the model, outside of known data. This is the test set of data. If acceptable ratings are achieved, the model is considered a success. If not, it indicates that changes need to be made, either regarding the data used, or the architecture of the model itself. Metrics used for this rating are commonly chosen from among accuracy, precision, recall, F1-score, or area under the Receiver Operating Characteristic (ROC) curve (AUROC).

While supervised learning remains dominant in computer vision, unsupervised and self-supervised approaches can be explored as alternatives. Unsupervised learning focuses on using unlabeled data, and works with the objective of discovering underlying and hidden patterns in the data, without any explicit guidance. Clustering is a very typical and common example of an unsupervised learning technique, where the program attempts to discover similarities between given examples, and group them according to their degree of similarity/dissimilarity. [38] Self-supervised learning works by creating auxiliary tasks to generate labels automatically, also enabling the use of unlabeled data. [39]

Supervised learning still occupies the spot of the most used method of training for pain assessment algorithms, with higher recorded levels of performance, as well as easier reproducibility. [8]

A problem that plagues all deep neural networks is overfitting, the phenomenon where the model, instead of learning broader patterns, memorizes the training data specifically, resulting in poor generalization. While some architectures are more prone to this than others, it is good practice to employ some techniques to counter this.

Data augmentation is a very common one, whereby the dataset is artificially increased through alterations made to the existing data, such as flipping, rotation, cropping or element obfuscation. This increases the range of inputs in the training phase, usually improving robustness. Dropout is another standard technique, where randomly selected neurons are simply ignored during training, with the hopes of preventing co-adaptation of features. Batch normalization is another widely applied technique. Here normalization is applied to the intermediate activations within the model, stabilizing training. In Transformers, a broader layer normalization is often chosen instead.

More traditional options include L1 and L2 regularization. They each add a penalty term to the objective function and control model complexity. L1's penalty encourages the sum of absolute values of the parameters to converge towards a minimum, while L2's penalty strives for the minimization of the square error, penalizing large weights.

Finally, early stopping is a simple measure that stops the training process before performance drops for the validation set, by trying to predict when the model would start overfitting. This also saves computational power in case training stops before the specified number of epochs.

These are the more standard practice options. More complex and specialized additions, like knowledge distillation, that includes the use of a larger "teacher" model to guide the training of the smaller "student" model. Instead of using the ground truths of the dataset directly, like the teacher model, the student utilizes the probability outputs of the teacher, that contain more information about class relations, to learn instead to mimic its behavior. [40]

While all these methods bring challenges on their own, the counter of overfitting makes them essential in modern practice.

Chapter 3

State-Of-The-Art

This chapter goes into detail through the state-of-the-art methods that exist for every step taken in this work.

3.1 Databases

3.1.1 Used Databases

This section describes the databases that were used on this exploratory work. The selection of these two databases was guided by their direct relevance to the research objectives, their availability at the time of analysis, and their established prominence within the community, both with documented results within the literature.

3.1.1.1 BioVid Heat Pain Database

The BioVid Heat Pain database [41] (Figure 3.1) was collected in a collaboration between the Universities of Magdeburg and Ulm. It contains data from 90 healthy adults, with ages between 20 and 65, subjected to experimentally induced heat pain at four intensity levels, adjusted to the pain threshold of each individual. Data collection was performed twice for each subject, once to focus solely on the face and facial expressions, and once with facial electromyogram (EMG) sensors. In addition, the participants were also asked to perform posed basic emotions and pain, and were stimulated with video clips to evoke other basic emotions. This database is thus divided into 5 parts.

- **Part A** corresponds to short time window pain stimulation videos without EMG electrodes on their face, along with some bio signals, namely galvanic skin response (GSR), electrocardiogram (ECG) and EMG at the trapezius muscle;
- **Part B** is similar to Part A, only now with videos of the subject faces occluded with electrodes, and additional EMG signals at the corrugator and zygomaticus muscle;

- **Part C** is the same as Part A, but with longer videos, and including all bio signals mentioned for Part A;
- **Part D** corresponds to the posed pain and basic emotions, and the corresponding bio signals, GSR, ECG and EMG at the trapezius muscle;
- **Part E** corresponds to the emotionally elicited videos, with all the associated biosignals mentioned in Part D.

The annotations of this database are only based on the calibrated stimulus for each individual. Part A contains data on only 87 of the 90 total subjects.

The combination of visual and bio signals makes this dataset a prime option for exploration of potential correlations between them, and the vast scope of subjects allows for either a widespread study, or the potential to pick and choose a specific age range or gender to be studies, while maintaining a respectable count of subject variability.

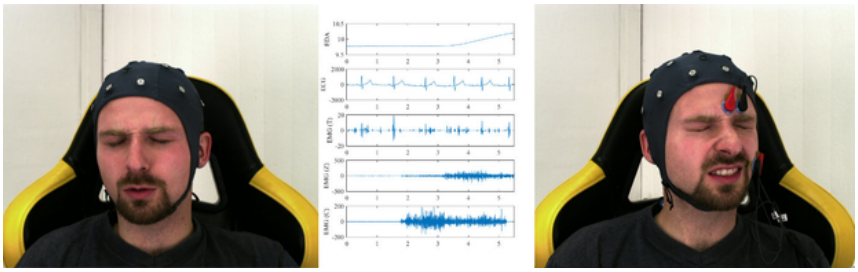


Figure 3.1: Example images and information of the BioVid Database. [41]

3.1.1.2 Multimodal Intensity Pain Database (MIntPAIN)

The Multimodal Intensity Pain Database (MIntPAIN) [42] has only 20 subjects, ages ranging from 22 to 42. In this case, pain was induced through electrical muscle pain stimulation, and similar to the BioVid database, participants were randomly exposed to 4 different intensities of stimuli. All participants were exposed to two trials, each with 40 sweeps of pain stimulation, and in each sweep, a label of "no pain" (Label0) or "pain" (Label1-Label4) was collected. Each trial resulted in 80 folders from these 40 sweeps, some missing for some subjects.

In addition to regular RGB video (Figure 3.2a), depth (Figure 3.2c) and thermal (Figure 3.2b) imaging videos were also recorded for each stimulation.

The annotation associated comes in the form of the stimulus intensity levels, which were calibrated on a per person basis, and self-reported levels of pain.

Although containing a small amount of subject variety, the multiple video modalities obtained may be interesting to explore more deeply the mechanisms of pain and correlations between those specific visual modalities.

(a) RGB (b) Thermal (c) Depth

Figure 3.2: Example images of the MIntPAIN Database. [42]

3.1.2 Not Used Databases

This section is dedicated some of the databases that were not used, but are nonetheless noteworthy.

3.1.2.1 UNBC-McMaster Shoulder Pain Expression Archive Database

The UNBC-McMaster Shoulder Pain Expression Archive Database [43] (Figure 3.3) is a collaborative project between the University of Northern British Columbia and McMaster University, designed to capture and analyse facial expressions related to genuine shoulder pain. It includes data from 25 adults aged 18 to 69, each experiencing shoulder pain and undergoing range-of-motion tests on both their affected and unaffected limbs. During these tests, participants' natural facial expressions of pain were recorded.

The database is composed of 200 video sequences, capturing spontaneous facial expressions tied to pain. Detailed FACS annotations are present across 48,398 frames, documenting specific facial muscle movements. Pain intensity is assessed through several measures: frame-by-frame pain scores, sequence-level self-reports, and observer pain intensity (OPI) ratings by trained evaluators. Finally additional data includes 66-point Active Appearance Model (AAM) landmarks for each frame to support the tracking and analysis of facial features.

Self-reported pain levels are based on Visual Analogue Scale (VAS) scores, with sensory and affective verbal ratings provided by participants, supplemented by observer pain intensity scores. Each sequence also notes which limb was tested, the affected or unaffected side.

This dataset is particularly valuable for advancing research in automatic pain detection and facial expression analysis within clinical settings. It includes a wide range of pain intensities and accommodates various head poses, although most images are near frontal. It is without a doubt the most influential and used dataset on the field. However, the database seems to no longer be publicly available upon request, meaning the only way to access it as of now is through contact with researches that have already used it, and potentially retain access to it.

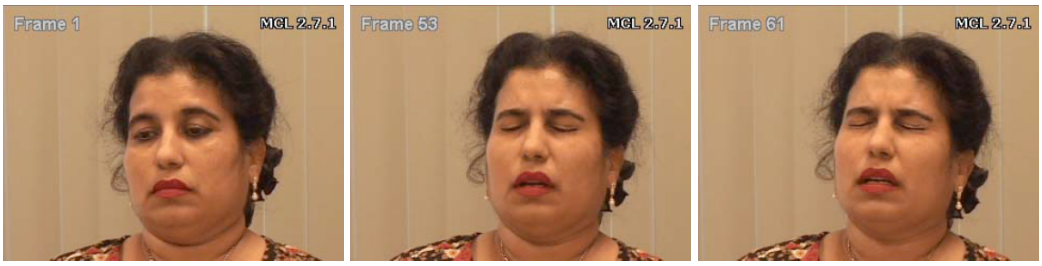


Figure 3.3: Example images of the UNBC-McMasters Database. [43]

3.1.2.2 EMOPAIN Database

One very recent, and influential database in this area is the EmoPain database, used for the EmoPain Challenge 2020 [44].

This was the first international competition with the objective of permitting a common platform for comparison of different approaches to chronic pain assessment derived from expressive components, and the classification of said components. The initiative was created with the objective of addressing one of the bigger problems related to the area, which was the lack of publicly available datasets able to be properly used for training of predictive systems with multimodal approaches. It consists of audiovisual, motion data and muscle activity captured from chronic lower back pain and healthy participants, engaged in everyday-life replicating scenarios and activities. The face dataset comprised 36 participants, being partitioned randomly into a training set, with 8 chronic lower back pain and 11 healthy participants; a validation set, with 3 chronic lower back pain and 6 healthy participants; and a test set, with 38 chronic lower back pain and 5 healthy participants. The movement challenge dataset comprised 30 participants, being divided in a similar way to the face dataset.

The challenge presented three distinct recognition tasks: (i) Pain Estimation from Facial Expressions Task, (ii) Pain Recognition from Multimodal Movement Task and (iii) Multimodal Movement Behaviour Classification Task, of which participants could choose one or multiple to compete in. The papers accompanying this challenge were presented at the FG2020 International Workshop on Automated Assessment of Pain.

The EmoPain Challenge 2020 provided a valuable platform to test and compare methods, paving the way for further developments in automated pain assessment and multimodal analysis. Unfortunately, the database is no longer available on the official platforms of the challenge, though another similar event would be beneficial to propel once again research in this area.

3.1.3 Overall Landscape and Comparison

The Table 3.1 presents an overview of some of the most commonly used and more recent pain-based datasets, as well as their specifications in terms of modalities, size, and type of subjects. As can be seen, these databases vary considerably in both the number of subjects and types of modalities included. The only constant modality is video, establishing it as the most accessible and widely applicable one when the objective is the development of an user-friendly model. Notably,

only two datasets from the ones listed actually include instances of naturally occurring pain, none of which were available for use. Although it was elicited by forced movement, these datasets stand out from the rest, where pain was artificially induced by external stimulation. Given the variability and limitations of available datasets, effective preprocessing techniques are crucial to ensure comparability and to maximize their utility.

Database	Sensor Modalities						Stimulus Modalities						Subjects	Population/Condition
	Video	Audio	EMG	ECG	EDA	Other	Thermal	Electrical	Pressure	Movement	Phasic	Tonic		
UNBC-McMaster [43]	x									x			25	Shoulder Pain
BioVid [41]	x		x	x	x		x				x		90	Healthy
BP4D+ [45]	x			x	x	x	x					x	140	Healthy
EmoPain [44]	x	x	x			x				x			30	Chronic Low Back Pain
SenseEmotion [46]	x	x	x	x	x	x	x				x		45	Healthy
MIntPAIN [42]	x					x		x			x		20	Healthy
X-ITE [47]	x	x	x	x	x	x	x	x			x	x	134	Healthy
PEMF [48]	x						x		x		x		68	Healthy
AI4Pain [49]	x					x		x			x		65	Healthy
SynPAIN [50]	x												Synthetic	Generated

Table 3.1: Comparison of pain-related databases. “x” indicates the presence of a given modality. (Phasic - short, externally induced pain; Tonic - longer, sustained pain.)

3.2 Facial Pain Recognition

This section details the current thought process behind most automatic pain assessment facial emotion recognition methods. These methods can broadly be divided into three big steps: Facial Detection and Preprocessing; Feature Extraction; and Pain Assessment.

3.2.1 Face Detection and Preprocessing

Preprocessing in pain detection can occur at different levels of complexity, depending on the features to be extracted, the classifier structure, and the dataset characteristics.

When considering just the information contained within single isolated frames or images, the first step for almost all algorithms in this area is to locate the face within the utilized footage. While not strictly necessary in most cases, due to the databases used being usually captured in controlled environments specifically for the facial detection purpose, or the fact that some architectures are already very efficient even without it, it would be a crucial step to take if one intends to adapt the process to be applied to live footage. In these situations, the target subject’s face may not be fully aligned, or in an optimal position for feature extraction. Additionally, locating the face and limiting the presence of all the otherwise useless information from the image or video also helps the model be more cost effective, reducing the amount of useless information fed into the feature extraction stage.

Around the 2000s, the algorithm most used for this purpose was the Viola&Jones. [51] For several years, it was considered state-of-the-art and remains in use today, though it’s now seen as less efficient compared to more modern, complex methods. It also paved the way for future Haar Cascade classifiers, named for their resemblance to Haar wavelets. These classifiers are

effective at identifying specific structures in images, such as edges and texture changes, making them valuable for facial recognition tasks, much like their predecessor.

A more robust approach is the combined use of Histogram of Oriented Gradients (HOG) and Support Vector Machines (SVM). [52] In this method, HOG functions as a feature descriptor, detecting the shape and structure of objects in an image by analyzing gradient orientations and grouping them into "bins" across smaller image sections, much like histograms. These sections are later normalized into larger groups to mitigate variations in color and brightness, producing a feature vector that summarizes the local gradients across the image. This vector is then processed by an SVM, a supervised machine learning technique commonly used for classification and regression tasks, which effectively separates data into distinct classes. In this context, SVM processes the feature vector from HOG analysis, and optimally classifies it to distinguish faces from other elements in the image.

Earlier face detection algorithms, such as Viola&Jones and HOG-SVM, laid the groundwork for modern approaches. One of these prominent deep learning techniques is the Multi-Task Cascaded Convolutional Network (MTCNN), a tool introduced in 2017, that stayed for several years as the benchmark for facial detection.

MTCNN was developed to overcome the limitations such as sensitivity to pose, scale, lighting variations, and misalignment of faces, while combining detection and landmark localization. Although newer alternatives have been cropping up, MTCNN can still be used as a teaching tool, and found to be used in recent papers, emphasizing its previous prominence. [53, 54, 55, 56, 57]

MTCNN integrates facial detection and alignment within a single framework by using a cascade of three Convolutional Neural Networks (CNNs): the Proposal Network (P-Net), the Refinement Network (R-Net), and the Output Network (O-Net). The P-Net works as a primary identifier of potential faces. It applies a series of convolutional filters to the original image, generating bounding boxes that may contain faces. Next, the R-Net refines these candidate bounding boxes by cropping and standardizing their size. These regions are then processed through additional convolutional and fully connected layers to confirm the absence or presence of a face. Finally, the O-Net further refines the bounding boxes and detects specific facial landmarks, refining their location. This is achieved by taking the boxes identified by the R-Net as containing faces and again cropping and resizing them, and passing them through another section of convolutional and fully connected layers. This is done to enhance the accuracy of both face and landmark final localization.

As per the natural cycle of progression, over time MTCNN was outdone by more recent options. Its compatibility with older and shallower models contributed to its continued use even after better performing models were introduced. Recently it has become less and less desirable, although still serviceable. For pain datasets, its resistance to misalignment made it very useful for when subjects moved slightly during recordings.

In 2019, Jiankang Deng introduced RetinaFace [58], a competitor to MTCNN and one of the current in consideration state-of-the-art options for facial detection. RetinaFace was originally

developed to address some of MTCNN's limitations, particularly in detecting faces under challenging conditions such as extreme poses, occlusions, and small-scale faces. On benchmark tests, for example the WIDER FACE hard test set, RetinaFace outperformed the best models at the time, achieving higher accuracy and more robust landmark localization. Specifically, it was able to reach an Average Precision of 91.4%, about 1.1% above the previous state-of-the-art. A major distinction is RetinaFace's ability of locating 3D facial landmarks. While the base MTCNN is limited to detecting 2D landmarks, RetinaFace works in 3D, being able to better handle occlusions and estimate facial geometry based on head positioning, improving detection and landmark localization under complex circumstances.

Not unlike MTCNN, RetinaFace integrates both facial detection and landmark localization in a single framework. It employs a single stage, anchor-based design enhanced by use of a feature pyramid network [59]. This network combines multiple convolutional layers to detect objects on different scales, enhancing the ability of the model to detect smaller faces, or just in general multiple faces in crowded images, at little increase to computational cost. RetinaFace uses a multi-task loss function to simultaneously try to predict face classification (the probability of an anchor being a face), face box (the coordinates of the potential face), the five traditional facial landmarks (two eyes, two corners of the mouth and nose) and the 3D facial geometry and rough location of the facial pixels, all while allowing for real-time detection using few resources.

Despite its strong performance in computer vision in general, it is still rare to find models taking advantage of RetinaFace in pain-related works, which most often rely on MTCNN or derivatives of it. This may just come from conformation to the methods, or the lack of need for 3D alignment or scalable detection. Pain related databases are composed of individual videos, where the camera has a clear view of the subjects face. In this context, MTCNNs lighter design may actually make it more desirable, and the added complexity of RetinaFace may not justify its use as of yet.

After facial detection, several preprocessing steps can be applied to prepare it for feature extraction, enhancing the accuracy and efficiency of the analysis. These steps may include resizing, normalization, and alignment to ensure consistency across samples. [60, 61, 62] For instance, resizing standardizes the dimensions of each detected face to match the input size expected by the model, making the processing of each image uniform and reducing computational costs. Normalization is another crucial step, where the pixel values of the images are adjusted to account for variations in lighting and contrast. Normalizing these values can help the model focus on the shape and texture features of the face itself rather than on irrelevant fluctuations caused by lighting differences. Alignment is also frequently employed to ensure that the faces in each image have similar orientations. This can involve rotating or shifting the facial regions so that the eyes, nose, and mouth appear in consistent positions across samples.

An additional process that is frequently performed previously to feature extraction is data augmentation. [55, 63, 64, 65] This is an important step in ensuring the existence of enough training data for the algorithm, while also providing some variations to boost the generalization of the model. Image flipping and rotation are some of the more common and simple to perform

processes in this context, as they don't lead to warping of potentially useful facial features, and preserve facial structure while still creating usable variations.

In the case of pain detection, all well established databases have some form of video feature, and thus a natural and correct sequence of frames. An extra level of information can be extracted from the temporal evolution of the phenomenon. This can add some additional steps to the preprocessing phase. For instance, interval isolation has been explored in studies that attempt to analyze sequences in which pain expression is most intense or prominent. [66] Isolating these intervals allows the model to focus on moments when facial expressions related to pain are at their peak, augmenting the probability of capturing transient features associated with discomfort.

After faces are detected and prepared, the next stage involves extracting features, which can occur at different levels of abstraction depending on the model and task requirements.

3.2.2 Feature Extraction

In order to analyze a dataset, one must first choose which aspects are in fact relevant for the intended study. This is the step that can more easily vary between different models, depending on what front the developers decide to focus on, and can dictate what methods of preprocessing are employed. This is because, if the features extracted are good, then the resulting classification should also be improved. We can discern some general groups to which the features typically used in FER belong to, and that are common in the specialized space of pain recognition.

The two more universal classes are the geometric and appearance features. Geometric features are based on facial landmarks, for example the corners of the mouth and eyes, and in evaluating distances between key points, areas and shapes defined by landmarks, representing the facial geometry as a whole. These features are essential to capture the structural changes in facial expressions, such as the rising of eyebrows and widening of the mouth. While some databases already possess the annotations necessary to locate facial landmarks, when that is not the case there are some techniques that are commonly employed. Active Appearance Models (AAM) are one of these techniques. [64, 67, 68, 43]

Appearance features, on the other hand, capture texture and intensity information of the face. These include local texture descriptors like the previously discussed Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP), as well as global appearance features based on techniques like Principal Component Analysis (PCA). [61, 69] Appearance features are useful for detecting texture changes, like wrinkles and furrows that occur during expressions.

Many modern approaches combine both geometric and appearance features for improved performance, recognizing that each type of feature contributes unique information to the emotion recognition task.

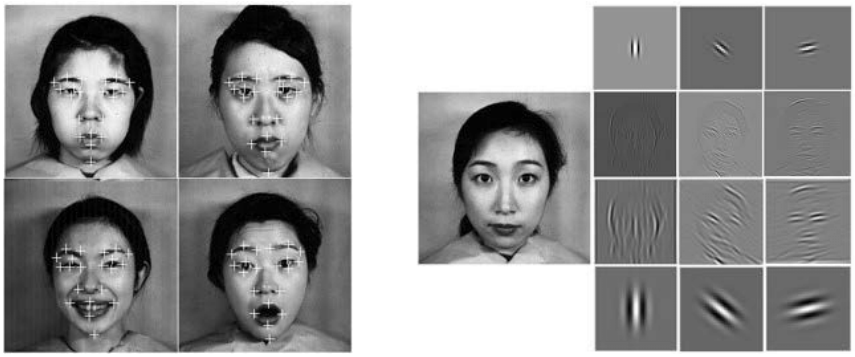


Figure 3.4: Geometry features (left) vs. Appearance features (right). [70]

Of special note for pain recognition in specific, are Facial Action Units (FAU's), and the associated Facial Action Coding System (FACS) [71] In FER based pain assessment, it has become the closest thing to a "universal" label. Using FACS for automatic pain assessment involves observing and assigning Action Units (AU's) to specific, predefined, thought to be pain-related, muscle movements that may indicate a presence or a certain level of pain. These can be considered as geometric features, as they rely heavily on the distances between facial landmarks for their discernment. Known for its reliability and objectivity, it is one of the most used annotation methods for pain related facial evaluations. It is limited to classifying clearly visible changes and movements, so more subtle variations, and phenomena unrelated to movement, like tears and excessive sweating, are disregarded and not utilized for consideration.

Other less standard or perhaps less easily defined feature groups also find their way into these works. As expressed before, temporal information is now being more explored as a source of potentially relevant information. Features extracted from the study of changes occurring over time are known as dynamic features, while those extracted from individual images are known as static features. An example of a dynamic feature is the Optical Flow Analysis (OFA) [72] This technique tracks movement between sequential frames, highlighting areas where motion is more pronounced, such as eyebrow raises or winces, which may correlate with pain levels. By incorporating temporal data through techniques like optical flow, the model can develop a richer understanding of expression changes, which can be essential for detecting and evaluating pain dynamics.

Finally, with the introduction of deep learning methods, a new category of features was also differentiated. These are called "Learned Features", as they are automatically and independently learned by the deep learning models, with the assumption that, even if the researcher does not know what they actually are, they will enable the model to reach accurate predictions. These are, as of the writing of this paper, considered state-of-the-art when relating to the creation of reliable automated FER algorithms.

One of the most common types of networks used for deep learning purposes are CNNs, and even more so when the data includes images.

While previous works would have to perform facial detection and feature extraction separately, nowadays there exist some open source and freely available architectures that can perform both

processes, depending on how widespread the use of the desired features is.

OpenFace [66, 73, 74, 75] is one that is regularly used in this field, particularly valued for its ability to automatically detect and track facial landmarks, estimate head poses, and recognize facial expressions and AUs based on FACS. Developed as an open-source software, OpenFace is specifically designed for ease of integration in research applications and is even able to be used for real-time acquisition and qualification.

According to [75] however, the average Pearson Correlation Coefficient (PCC) of the OpenFace 2.0 toolkit when applied to the DISFA dataset was of 0.59. While better than the alternative methods used in the study, and the current version now being 2.2, this still indicates an approximate 40% gap in accurate AU detection, which can serve to the detriment of the toolkit.

3.2.3 Pain Assessment

3.2.3.1 Traditional Assessment Methods

Even in more traditional approaches, pain classification can be acquired through various methods. These try to translate what a patient is feeling into a metric or language that can be interpreted by the caregiver. Due to the singular nature associated with the pain feeling, self-reporting and related metrics are considered the golden standard. There are several scales used to facilitate the self-reporting classification of pain. The three most widely used ones are numerical rating scales (NRSs), verbal rating scales (VRSs) and visual analogue scales (VASs). [8]

NRSs (Figure 3.5) are the simplest ones. They consist of a series of numbers that represent the range of pain intensity. Starting at 0, commonly they extend to 10, 20 or 100, with 0 corresponding to "no pain" and the chosen maximum to "most intense pain". This method lacks a meaningful wait to differentiate between two adjacent levels of pain, as the difference between two sets of adjacent values may not be the same, and is mostly used for its ease of employment.

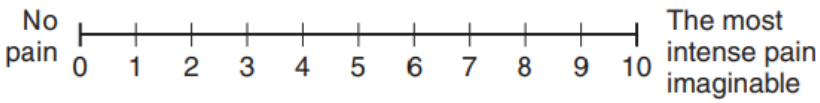


Figure 3.5: Example NRS.

VRSs (Figure 3.6) are a step up in complexity, using phrases to describe the pain intensity, ordered in an ascending fashion. As with NRSs, they cover the whole pain spectrum, from "no pain" to "excruciating pain". Multiple variations can be made from these models, depending mostly on the vocabulary of the developer of the scale, and the number of stages they desire to include. The pain stages can also be associated with numbers, to help with ease of interpretation. While their ease of comprehension and increased level of expression and adaptability to different and specific circumstances make them somewhat superior to NRSs, they suffer from the same weakness, the difference between two adjacent described stages may not be the same for all stages, again restricting model interpretability. Another limitation relates more to the patient, who needs

to be at least somewhat familiar with the language and expressions used, in order to accurately understand what they are communicating to the examiner.

None	0
Mild	1
Moderate	2
Severe	3
Very severe	4

Figure 3.6: Example VRS.

Finally, VASs (Figure 3.7) are simply made by a line, horizontal or vertical, working much like a ruler for pain, with verbal components at each extreme, representing the ends of the spectrum. The subject marks where in the line they consider their pain intensity to be located, not having to be equated to a corresponding number, although recent iterations of this method include a sliding marker superimposed on a VAS drawn on a ruler, to give some direction to both the patient and examiner. VASs are often recommended in place of the alternative, being able to maintain a more interpretable ratio and freedom of choice for the patient. However, it is also the more time-consuming method for the patient to respond to, potentially leading to some overthinking and biases to affect the final answer. Additionally, since they are scored based on the number of centimetres/millimetres, scoring time is also more time consuming.

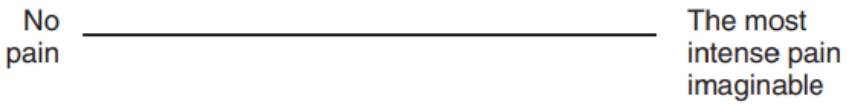


Figure 3.7: Example VAS

Self-reporting, although being the most reliable method currently available, is an imperfect method. It solely relies on the patient's ability to convey information through the chosen scale, without being affected by biases, both internal and external. However, it can be very useful for calibrating stimulation prior to data acquisition. [8]

A second option is the classification of pain by an external observer. This can be useful when none of the usual methods can be applied, because the subject is either non-communicative or has heightened difficulty in interpreting the scales. While they can be more objective, it requires the use of appropriate metrics to facilitate validation and comparison between studies and can still be influenced by human bias.

The previously mentioned FACS [71] is one such metric. As referenced in Section 2.1, it was created by Paul Ekman and Wallace V. Friesen with the objective of distinguishing all possible facial movements. It is also used as nomenclature when regarding facial movement research, and more importantly, in FER based pain assessment. Known for its reliability and objectivity, it is the most used annotation method for pain related facial evaluations. It is unfortunately limited to classifying visible changes and movements, so phenomena unrelated to movement, like tears, are disregarded and not considered for classification.

A very well-known system that utilizes FACS is the Prkachin and Solomon Pain Intensity (PSPI) metric. [76] It is a scoring system based on the FACS, with some self-report. The system tries to consolidate both internal and external perceptions of pain. While there are a total of 44 AUs specified by FACS, Prkachin and Solomon tried to focus on those that had already been speculated as corresponding to the existence of pain. These were: brow-lowering (AU 4), cheek-raising (AU 6), eyelid tightening (AU 7), nose wrinkling (AU 9), upper-lip raising (AU 10), oblique lip raising (AU 12), horizontal lip stretch (AU 20), lips parting (AU 25), jaw dropping (AU 26), mouth stretching (AU 27) and eye closing (AU 43). [76] The more commonly used equation to calculate the level of pain is as follows:

$$\text{PAIN} = \text{AU4} + (\text{AU6} \parallel \text{AU7}) + (\text{AU9} \parallel \text{AU10}) + \text{AU43}$$

There are scales created and medically validated to be used specifically for target populations. Neonates are evaluated according to NIPS [77], CRIES [78] and NFCS [79] for example, while for elderly people professionals rely on PACSLAC [80], PAINAD [81] or PADE [82]. Most of these consider multiple parameters, including facial expressions, but expanding to body language and bio signals.

3.2.3.2 Automatic Assessment Methods

When it comes to automatic assessment of pain, there are two big schools of thought. The more simplistic one, where the objective is solely to identify the presence of pain within the images, and the more complex one, where there is an attempt to categorize the levels of pain displayed. Additionally, some works also explore the option to differentiate between feigned and real pain, though this is a more niche avenue [83].

The scales used for the multiclass option are extremely varied. They can be arbitrarily created by the team themselves [62], calculated through analytical methods, such as Gaussian analysis [54], or try to conform to a general pain scale, such as the PSPI [76] or the Wong-Baker faces pain rating scale [57]. Since there are no guidelines, the chosen scale stems only from the objectives of the team. If the intention is comparison of performance with other models form the same database, then using the native annotation may be more advantageous, while trying to convert the results to a more used scale may help with experimentation across different datasets.

3.3 State-of-the-Art Models for Pain Recognition

Building on the methods described above, several types of models have been applied to automatic pain recognition. These range from classical machine learning approaches based on handcrafted features, to deep learning models that learn representations directly from data.

As one would expect, attempts to detect the presence of pain came first, and only after were made attempts to quantify it. And because of that, there are naturally more examples of "older"

methods being used for pain detection, as is the case for the multiple variations of SVMs and random forests (RFc).

S. Brahnam *et al.* [61] were probably some of the first to try this. The method proposed was initially a comparison between PCA, linear discriminant analysis (LDA) and SVM, but the SVM results far exceeded the other, becoming the focus of study. This was done in the COPE database, composed only of neonates.

Due to their popularity, certain archetypes of CNNs are commonly used as a starting point for multiple algorithms. Some examples are VGG16 [57, 65, 72] and VGG19 [57], ResNet18 [63, 72, 84] , ResNet50 [57, 63, 72, 84] and ResNet101V2 [57].

VGG16 and VGG19 are two early models known for their simple yet effective architecture. They consist of a sequence of convolutional layers, each followed by a rectified linear activation (ReLU), and occasionally by a max-pooling layer, which helps reduce dimensionality and retain important spatial information. VGG16 contains 16 weight layers (13 convolutional and 3 fully connected), and VGG19 contains 19 weight layers (16 convolutional and 3 fully connected). This straightforward design enables VGG models to learn detailed representations and achieve high accuracy on complex image classification tasks. However, they result in a large number of parameters, making them computationally expensive and memory intensive.

ResNet18, ResNet50 and ResNet101V2 are members of the Residual Network archetype, which introduced the concept of residual connections to address the vanishing gradient problem that affects most deep learning algorithms, allowing even deeper networks to be trained. These residual connections, or skip connections, allow gradients to bypass one or more layers, helping information flow efficiently through the network and enabling each layer to learn more complex patterns without the risk of performance degradation. ResNet18 is the shallowest one, with 18 layers, ResNet50 with 50 layers, and ResNet101V2 extends to 101 layers with some architecture improvements, offering even greater depth. The hallmark of ResNet architectures is their efficiency in training very deep networks, making them effective for tasks such as object detection and image classification.

Safaa El Morabit *et al.* [84] capitalized on the popularity of CNNs by thoroughly analyzing several well established CNN archetypes, namely MobileNet, GoogleNet, ResNeXt-50, ResNet18, and DenseNet-161, and their capabilities for both serving as feature extractors, and for directly classifying the pain. Their conclusion was that the approach with better results was indeed to use them as feature extractors, and feeding the output feature vectors to other dedicated classifiers that would use them to learn. They however relent that these conclusions are dataset and model restricted.

Mohammad Tavakolian *et al.* [72] put a spin on this by attempting to train the models through self supervision. By taking multiple well established CNNs, and using a Statistical Spatiotemporal Distillation (SSD) to create singular RGB maps for each video, and using those to train the models, they were able to not only permit 2D-CNNs to process video, but achieve good results al well.

Xuwu Xin *et al.* [64] escaped this by proposing a brand new type of of deep learning framework entirely, the Locality and Identity Aware Network (LIAN). This came from struggle of reg-

ular models identify the subtle, localized facial movements that signify pain, and that vary significantly across individuals. LIAN addresses these challenges with two complementary modules, Locality-aware and Identity-aware. The former guides the network to focus on the most informative facial regions by combining feature extraction with an attention mechanism. The latter uses a multi-task learning approach to model subject-specific traits. Together, these modules help achieve identity-invariant yet locality-sensitive pain recognition.

Even so, CNNs are not the only type of deep network used, even if they are by far the more common ones. Other experimented with options include Long Short-Term Memory (LSTM) networks [65], Transformers [56] , and, more recently, Radial Basis Function Based Extreme Learning Machines (RBF-ELM) [85].

LSTMs are a type of recurrent neural network (RNN) designed to learn long-range dependencies by utilizing memory cells, exemplified by Pau Rodriguez *et al.*. These cells enable LSTMs to retain information over extended periods, making them well-suited for analysing sequences where context matters. In pain detection, LSTMs can effectively capture how expressions change from frame to frame and can be used when wanting to account for the temporal progression of events. [65]

Transformers, as per example the one used by Safaa El Morabit and Atika Rivenq, originally developed for natural language processing, have also found some application. They leverage self-attention mechanisms to weigh the importance of different parts of the input data, allowing for a more flexible approach to learning relationships within the data. By processing data in parallel rather than sequentially, transformers can handle larger datasets efficiently and can capture complex dependencies between features, which is particularly beneficial when dealing with high-dimensional image data. [56]

More recently, Radial Basis Function Based Extreme Learning Machines (RBF-ELM) have emerged as an alternative approach, as is the example of the network developed by Sherly Alphonse and S. Abinaya and Nishant Kumar. RBF-ELM is a type of single-layer feedforward neural network that uses radial basis functions as activation functions. This architecture allows for faster training compared to traditional neural networks, as it does not require iterative optimization of weights. RBF-ELMs can quickly approximate complex functions, making them suitable for tasks that require real-time processing, such as pain assessment from video feeds. [85]

The following Tables 3.2 and 3.3 summarize automatic pain detection approaches, focusing only on those that use only visual information or features extracted from images and videos.

Table 3.2: Summary of representative models for binary pain recognition.

Type	Model	Dataset (Pain)	Year	Metric	Performance
ML	SVM [61] (polynomial, linear)	COPE	2006	Accuracy	87.4%, 82.4%
	SVM [68] (linear)	UNBC-McMaster	2009	Hit Rate	82.4%
	SVM [67] (linear)	UNBC-McMaster	2009	AUROC	78.37
	SVM [43] (LIBSVM)	UNBC-McMaster	2011	AUROC	83.9
	RFc [73]	BioVid, X-ITE	2019	Accuracy	70.8%
	SVM [74] (linear)	UNBC-McMaster, BioVid	2019	Accuracy	88.4%
DL	NN [60]	UNBC-Physiology Lab	2006	Accuracy	91.67%
	VGG16, LSTM [65]	UNBC-McMaster	2017	AUROC	91.3, 93.3
	MobileNetV2 [73]	BioVid, X-ITE	2019	Accuracy	72.6%
	CNN [74]	UNBC-McMaster, BioVid	2019	Accuracy	99%
	CNN [86]	UNBC-McMaster	2020	Precision	99%
	DeiT [56]	UNBC-McMaster, BioVid	2022	Accuracy	84.15%, 72.11%
	Multiple CNN based models [57]	CP-PAIN, UNBC-McMaster, MIntPAIN, Delaware	2024	Inter-class Correlation Coefficient	0.751

Table 3.3: Summary of representative models for multiclass pain recognition.

Type	Model	Dataset	Year	Metric	Performance
ML	SVM (RVM) [62]	COPE	2010	k-coefficient	0.79
	SVM (non-linear) [87]	Hi4D-ADSIP	2012	Accuracy	92.9%
DL	ResNet18,ResNet50 [63]	BioVid	2020	Accuracy	36.6%, 90%
	CNN + TCN [69]	UNBC-McMaster, MIntPAIN	2020	Accuracy	94.14%, 89%
	SSD + CNN [72]	UNBC-McMaster, BioVid	2020	AUROC	69.2, 65.5
	S-PANET (CNN) [55]	Private	2021	Accuracy	97%
	LIAN [64]	UNBC-McMaster, BioVid	2021	Accuracy	89.17%
	MobileNet_v2, GoogleNet, ResNeXt50, ResNet18, DenseNet-161 [84]	UNBC-McMaster	2021	MSE	0.69, 0.64, 0.68, 0.80, 0.54
	RBF-ELM [85]	UNBC-McMaster	2024	Accuracy	99.1%

Chapter 4

Methodologies

This chapter explores in detail the methods implemented within the scope of the dissertation. While the present chapter outlines only the conceptual and technical approaches, Chapter 5 expands on how these methods were concretely applied in practice.

4.1 Evaluation Framework

This section describes the evaluation framework defined to train and evaluate the created models. As is typical for model training, the data was divided into three different disjointed subsets: Training, Validation and Testing.

The training subset and phase is generally the biggest and most time-consuming one. It consists of feeding a major part of the dataset to the model in order to train it. This process involves dividing the training data into smaller chunks called batches, which are passed through the model in sequence. For each batch, the model performs a forward pass, where it makes predictions based on its current internal parameters. It then compares those predictions to the actual labels using a loss function, which quantifies how far off the predictions are.

Following that, the model performs a backward pass (also known as backpropagation), where it calculates how each parameter contributed to the error and adjusts the weights accordingly using an optimizer. This process of forward pass, loss computation, and backward pass is repeated for every batch in the training set.

Once all batches have been processed, the model has completed one epoch, also known as a full pass through the training data. This process is repeated for whatever number of epochs is deemed necessary, allowing the model to gradually improve its accuracy and minimize error by repeatedly refining its parameters.

At the end of each training epoch, follows a validation phase. In this, a smaller portion of the data, not included in the training step, is fed to the model, in order to analyze its performance on unseen samples. This enables hyper parameter adjustment and combats overfitting, by ensuring that improvements and changes are generalized beyond the training set.

Finally, the testing phase. Once training is completed, the final portion of data, containing exclusively unseen samples, is used to determine the final performance of the model. This provides as unbiased an evaluation as possible, to determine the generalization of the final model created. It is the closest simulation to how it would work in real world scenarios.

In our case, this data can be boiled down to a variety of images associated with specific annotations related to the dataset from which they were extracted. The ultimate goal was to make the model as robust as possible to unfamiliar data, and therefore able to adapt to real world scenarios. The process through which these data were prepared is better detailed in Section 4.2.

4.2 Dataset Preparation

The methods herein described were evaluated in the following databases: FER2013+ (for baseline comparison), BioVid Part A, and MIntPAIN (RGB subset). Since the pain datasets themselves are described in detail in Sections 3.1.1.1 and 3.1.1.2 respectively, this section focuses on the preprocessing and partitioning steps applied.

For FER2013+, the data corresponds to a variety of unrelated images, each associated with one of the six chosen basic emotions. Its predefined splits, Training, Public Test, and Private Test, were preserved for the purposes of experimentation.

In the case of the BioVid and MIntPAIN datasets however, the data is more complex. They consist of videos and frames of the same 87 and 20 subjects respectively, exposed to controlled pain stimuli. Each data point is associated with a labeled level of pain (0 through 4) determined by the intensity of the stimulus.

For binary classification, only samples from the *no pain* label 0 and *maximum pain* label 4 were retained to maximize contrast.

For BioVid, an equal number of videos per label was selected for each subject, and frames were extracted from the central portion of each video, to attempt and use only the height of response to the stimuli.

For MIntPAIN, since the RGB portion of the data was already available as annotated frames per video, and a differing number of frames for each video, an equal number of label 4 and label 0 frames from each video was used, corresponding to the lowest number.

Selected frames were then passed through facial recognition software, where faces were isolated using a bounding box of 256×256 and converted to grayscale. The final resolution of the images depended on the model being trained. For ResNet-based models, all inputs were resized to 48×48, ensuring compatibility with FER2013+, which was also used as a pretraining baseline. For the transformer-based DeiT model, all inputs were resized to 224×224, in order to permit initialization from ImageNet-pretrained weights, which require higher-resolution inputs.

4.2.1 Partitioning

For the BioVid and MIntPAIN datasets, to preserve the integrity of the training, validation and testing groups, partitions were applied not to the number of total samples, but to the number

of subjects, meaning images of a given subject could only be found in the partition associated with them. Due to the varying number of samples collected per subject, partition percentages may not correspond exactly to the intended value. To combat imbalance of representation of different classes within the datasets, the steps taken in dataset preparation ensured that each class had roughly the same percentage of samples within each partition.

For the experiments made within the scope of this dissertation, the available datasets were divided using a 80/20 partition, meaning 80% of any given dataset was used for training and validation (60% and 20% respectively), and 20% for testing.

4.2.2 Frame Extraction

The BioVid dataset required frame extraction, as it is the only one composed solely of videos and not images or frames. As such, to be able to process the database, frames had to be extracted. This, however, opened up the possibility to, instead of just processing the videos in their entirety, select the sections deemed more important or with more information.

For the purposes of this work, only Part A was used, being, as described in Section 3.1.1.1, comprised of 5.5 second recordings. To maximize the probability of capturing peak responses, frames were selected uniformly from within the middle 50% of the videos. A maximum of 10 frames were acquired from each video.

4.2.3 Facial Detection

As mentioned previously in Section 3.2, one of the first steps in this type of processes is facial detection. While not the pinnacle that it once was, MTCNN still presents itself as an extremely accurate face detector that is widely used, and as such, it was the algorithm adopted for this task.

It is of note that no facial detection was necessary for FER2013+, as it already consists of focused 48×48 images of faces, which were used directly as model inputs without resizing. With the remaining datasets, this process becomes invaluable, to limit the amount of unrelated content that makes it inside the models.

In the BioVid dataset, the facial detection and frame extraction processes were linked. As frames were selected to be utilized, MTCNN was used to find the face within. If no face was detected, an alternative frame within the same extraction interval was selected for examination. The process repeated until an eligible frame was found, all the frames from the interval were inspected, or the maximum number of frames were selected. The algorithm then moved on to the next video sample.

In the MIntPAIN dataset, since it was already comprised of different annotated frames, only facial detection was needed. MTCNN was applied to all available frames.

This step ensured that all samples used for training and evaluation contained only clearly detected faces, minimizing noise from background content.

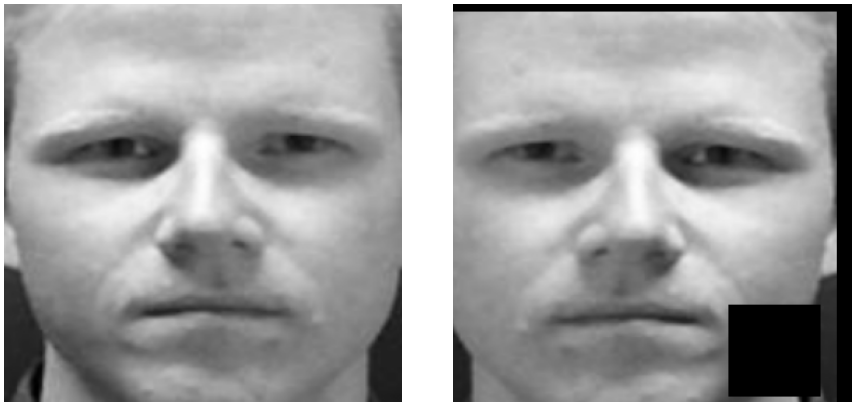
4.3 Data Augmentation Techniques

In addition to the regular data, some data augmentation techniques were implemented to increase variability and reduce overfitting. These include some standard alterations, such as small rotations and vertical and horizontal shifts.

On top of those, a chance to experience a complete horizontal flip and random occlusions were applied. These were not intended to simulate specific real-world scenarios, but simply to increase robustness against missing information. This all with the objective to diversify even more the datasets and extend the model's potential for generalization. The objective was to not fundamentally alter the input, at least not enough to change expressions too fundamentally.

All augmentations were applied post-cropping and resizing and prior to feeding them to the models.

Figure 4.1 shows these changes applied to an image from the MIntPAIN database.



(a) Original Image. (b) Augmented image.

Figure 4.1: Example of augmentation application.

4.4 Models

4.4.1 Altered ResNet-18

The primary model used for this dissertation is a slightly altered version of a traditional ResNet-18 architecture. It has been adapted for efficiency and reduced computational complexity.

Notably, the filter sizes were scaled down (for example, 24-192 instead of 64-512), and a smaller initial convolution kernel was used (3×3 instead of 7×7), making it more suitable to smaller datasets and constrained environments. The core concept, residual learning, was preserved through identity and projection shortcuts, and the residual blocks maintained the two-layer convolutional structure of the original design. This provided a good balance between depth and computational efficiency, all while benefiting from residual learning. Figure 4.2 shows a schematic the model in question. All input images for this model were standardized to a resolution of 48×48, consistent with FER2013+ and its pretraining setup.

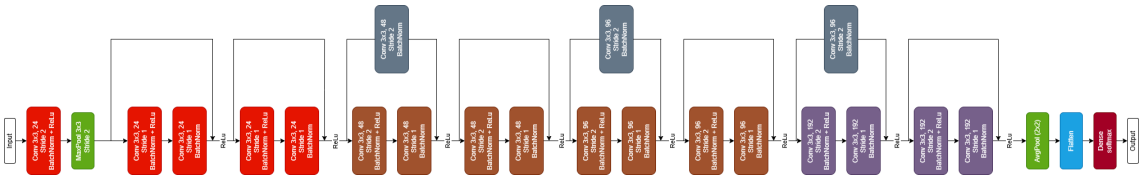


Figure 4.2: ResNet 18 inspired model.

4.4.2 ResNet-18 with Action Units (AUs)

To better incorporate both visual and semantic cues, a more complex architecture was developed by extending the altered ResNet-18 with an additional pathway for Action Unit (AU) features. The convolutional stream is identical to the model described in Section 4.4.1, providing a compact yet effective representation of the visual domain.

In parallel, a second input branch processes AU vectors. This pathway consists of two fully connected layers of 64 and 32 units, respectively, with ReLU activation, aiming to learn high-level embeddings of the AU information. The outputs of the image and AU branches are then concatenated and passed through a joint fusion layer of 128 units, before the final softmax classification layer.

The AU features to be fed into the second input branch were obtained via OpenFace, a process better explained in Chapter 5.

This design allows for the combination of pixel-level residual learning with interpretable AU-based descriptors, resulting in a richer representation of facial expressions. Figure 4.3 illustrates the dual-input model. Similar to the original version, the visual stream operated on 48×48 resized images, to maintain compatibility with FER2013+ initialization. The input vector contained 17 different AUs to be used for processing.

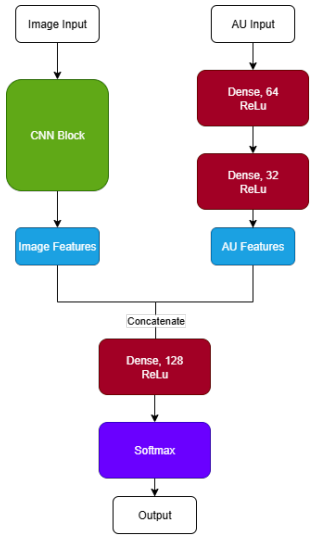


Figure 4.3: ResNet 18 with parallel branch for AUs.

4.4.3 DeiT-based Pain Detection

In addition these convolution-based models, a Transformer-based architecture was explored, leveraging recent advances in vision transformers. Specifically, the DeiT-Small distilled variant [88], used in [56] was employed, initialized with ImageNet-pretrained weights and fine-tuned for binary pain detection.

DeiT (Data-efficient Image Transformer) replaces convolutional feature extraction with a pure transformer encoder, operating on 16×16 image patches. The distilled variant uses knowledge distillation from a teacher network, improving generalization when trained on smaller datasets. In this dissertation, the DeiT backbone was adapted by replacing its classification head, instead mapping to the required number of classes for binary classification.

This approach enables a direct comparison between convolutional residual learning and transformer-based feature extraction. Figure 4.4 provides a schematic overview of the DeiT-Small distilled model. In this case, all images were resized to 224×224, to comply with the patch-based transformer input scheme.

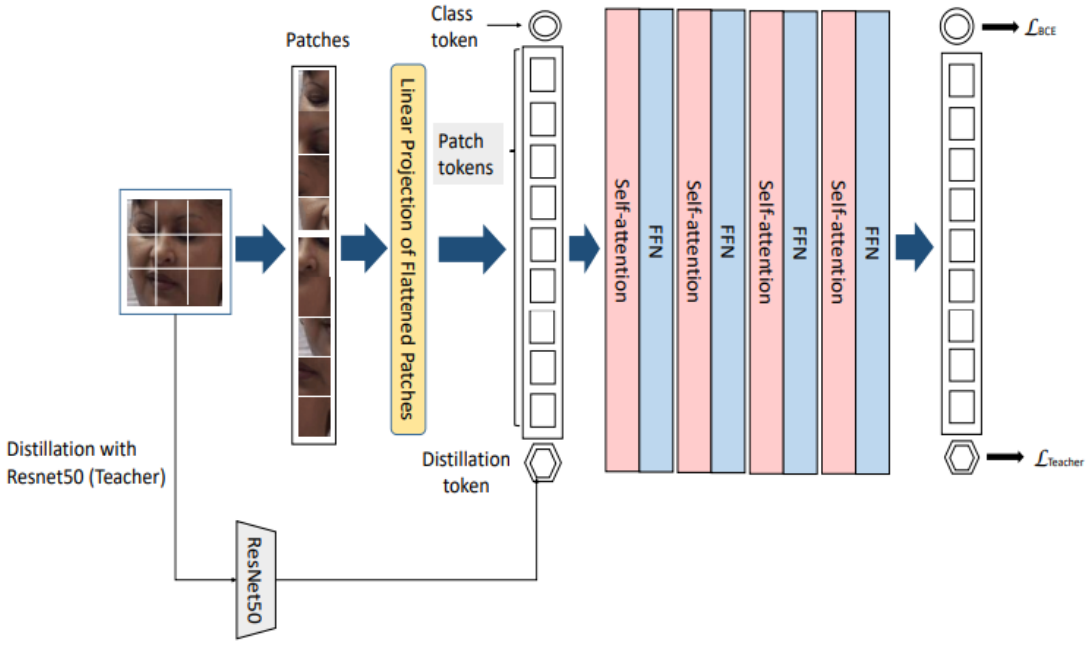


Figure 4.4: Data-efficient Image Transformer. [56]

4.5 Summary

The methodology adopted in this dissertation pertains to explore the differences between traditional emotion-based datasets and pain-specific ones, as well as the different performances we can expect in those datasets.

All data preparation and preprocessing centered around maximizing usable facial information, with MTCNN applied to isolate faces in BioVid and MIntPAIN, while FER2013+ was used directly. ResNet-based models operated on 48×48 grayscale inputs, to ensure compatibility with FER2013+ initialization, whereas the DeiT transformer relied on 224×224 inputs to enable ImageNet-based pretraining.

Regarding models, three main variants were implemented: ResNet-18 inspired model adapted for efficiency on small inputs, an extended version incorporating Action Unit features in a dual-branch architecture, and a transformer-based DeiT-Small distilled model, with a contrasting approach to feature extraction.

This methodological design enables a systematic comparison across datasets and architectures, assessing the viability of AFER approaches for pain detection in controlled scenarios.

Chapter 5

Experimental Work

The work developed for this dissertation sought to compare the performance difference of the implemented algorithms between traditional emotion-based datasets and pain-based datasets, experimenting with a shallow ResNet18 inspired model, and comparing it's performance to a more complex Vision-Transformer in the pain-based datasets.

This chapter details the experimentation process. A more detailed look into the evaluation and implementation steps taken can be found in Chapter 4.

5.1 Implementation Details

To be as forthcoming as possible, this section details technical specifications of the development and execution of the experiments. All experiments were conducted in a Python 3.12.1 environment. Code development and small-scale debugging were performed on a Windows system using Visual Studio Code, while large-scale training and evaluation were executed on the Linux-based SLURM cluster at INESC TEC, using the available Graphical Processing Units (GPU) resources. The ResNet18 models were trained using the partition with 11GB of RAM, while the DeiT was trained using the partition with 24GB of RAM.

The core libraries used in this dissertation include TensorFlow/Keras 2.19.0 for the ResNet18 implementations, and PyTorch 2.8.0 for the DeiT model, implemented with methods analogous to those available in Keras.

OpenCV 4.11 was employed for data preparation, including resizing and augmentation. Finally, for AU extraction, OpenFace 2.2.0. was employed.

For results evaluation and visualization, scikit-learn 1.6.1, seaborn, and matplotlib were used. Additional Python libraries such as NumPy and pandas supported data handling and preprocessing.

5.2 Dataset Realization

As per Section 4.2, the datasets were prepared for use. FER2013+ was left untouched, while BioVid and MIntPAIN went through the processes described.

The frame extraction process resulted in a total of 39,087 frames being analyzed from the BioVid dataset, and 13,688 from MIntPAIN. However, the employed method of facial detection, MTCNN, was not able to recognize faces in every single frame. This speaks to the still inconsistencies of this method. In total, only 34,394 frames made it to the final BioVid datset, and 12,524 to the MIntPAIN. Since it is known that every single frame should've resulted in positive detection from MTCNN, we can surmise that it had an accuracy of 87.99% on BioVid, and 91.50% on MIntPAIN. This imperfect detection may stem from various factors. Subjects in the BioVid dataset are all wearing EEG caps, and wrongful adjustments of these may have lead to difficulty in identifying the face. Additionally, other subjects whose faces the algorithm has difficulty locating show a pattern of possessing thick facial hair, obscuring the mouth and therefore two of the facial landmarks the algorithm relies on. For MIntPAIN, subjects are more distant from the camera, so it is more likely that their features are not correctly detected.

Partition was actually made before frame extraction, at a subject level. For the BioVid database, 52 were used for training, 17 for validation and 18 for testing. For MIntPAIN, 12 for training, and 4 for validation and testing each. Tables 5.1 and 5.2 list the subjects associated with each group, and Tables 5.3 and 5.4 display the final count os samples in each dataset per label and partition.

The 256×256 face bounding boxes were stored as greyscale values on a *.csv* file, one for each dataset, each with additional information, including the label associated with them (0-no pain, 1-pain), the subject they came from, guides to the specific video or frame they were extracted from from, and the partition they belonged to.

Table 5.1: Subject distribution across partitions for the BioVid dataset.

Training Subjects	**Validation Subjects**	**Testing Subjects**
080609, 100417, 080709, 101609, 112016, 083009, 082414, 083013, 072609, 102514, 101209, 081714, 101916, 091809, 112909, 082315, 100117, 083109, 080309, 081609, 102316, 101216, 072714, 111609, 082109, 120614, 092514, 112209, 080314, 092808, 071814, 111409, 092509, 110909, 082714, 092014, 081617, 112809, 112009, 080714, 092714, 100514, 120514, 071911, 101514, 110810, 082814, 100509, 072414, 112914, 101908, 102008	102414, 111313, 080209, 092813, 101809, 082909, 071313, 071309, 102214, 111914, 091914, 101015, 081014, 100909, 091814, 071709, 082208	082014, 072514, 101309, 101814, 083114, 102309, 100914, 080614, 110614, 092009, 073114, 073109, 112610, 112310, 100214, 082809, 101114, 071614

Table 5.2: Subject distribution across partitions for the MIntPAIN dataset.

Training Subjects	**Validation Subjects**	**Testing Subjects**
Sub3, Sub17, Sub4, Sub18, Sub14, Sub1, Sub15, Sub12, Sub16, Sub19, Sub9, Sub7	Sub5, Sub20, Sub11, Sub10	Sub2, Sub6, Sub8, Sub13

Table 5.3: Frame distribution for the BioVid dataset.

Label	Training	Validation	Testing	Total
0 (No pain)	10,176	3,400	3,600	17,176
4 (Max pain)	10,226	3,392	3,600	17,218
Total	**20,402**	**6,792**	**7,200**	**34,394**

Table 5.4: Frame distribution for the MIntPAIN dataset.

Label	Training	Validation	Testing	Total
0 (No pain)	3,995	1,208	1,046	6,249
4 (Max pain)	4,003	1,180	1,092	6,275
Total	**7,998**	**2,388**	**2,138**	**12,524**

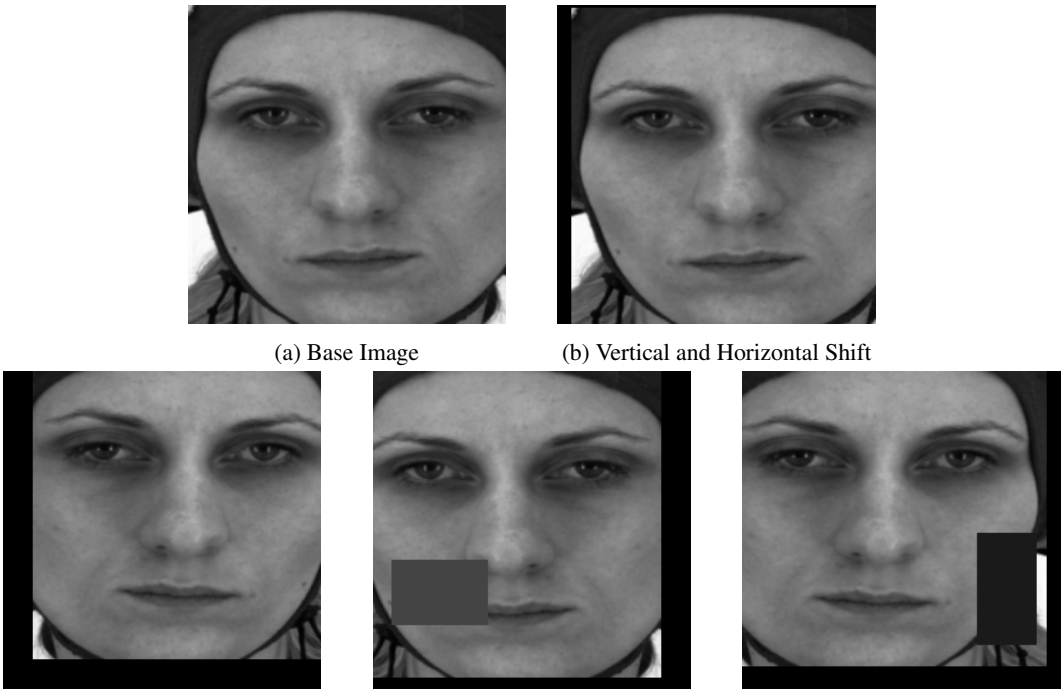
5.3 AU Extraction

The extraction of the Facial Action Units was then performed in the already stored faces, as to make sure both models used the exact same datasets. The faces were rebuilt from the stored grayscale values, and OpenFace's native "FeatureExtraction" command was used to automatically extract gaze features, eye landmarks, facial landmarks, head pose, and more importantly, Facial Action Units. Of these, thirty five were extracted, as they were the ones directly related to FAUs. Eighteen are related to the presence or absence of a given AU, and seventeen to the intensity of it. For the purposes of this work, only the ones representing the intensity were used. These were AU01_r, AU02_r, AU04_r, AU05_r, AU06_r, AU07_r, AU09_r, AU10_r, AU12_r, AU14_r, AU15_r, AU17_r, AU20_r, AU23_r, AU25_r, AU26_r and AU45_r, all of which are identified in Figure 2.1.

5.4 Implementation of Data Augmentation

Instead of applying all data augmentation techniques to all models, it was decided that exploring the effect the most noteworthy augmentations could have on the models would be worthwhile. Besides minor vertical and horizontal shifts of at maximum 10% of the original image size, and rotations of, at a maximum, 10 degrees, the changes deemed more drastic were the added probability of a complete horizontal shift; and the addition of random occlusion. As such, in regards to augmentations, models had four iterations. One with only the base augmentations, one with either of the major ones, and one with both augmentation methods applied. Horizontal Flip simply added a 50% probability of the image being completely flipped on the vertical axis, effectively becoming mirrored. Random occlusion was adapted from [89]. This occlusion would affect a maximum of 10% and a minimum of 2% of the image area. While the horizontal flip was included

to simulate natural variation to the data, random occlusion was included in the hopes of giving the models more robustness in non controlled environments. Figure 5.1 presents an original image and multiple of the potential variations applied to it during training.



(a) Base Image (b) Vertical and Horizontal Shift

(c) Horizontal Flip (d) Random Occlusion (e) Flip + Occlusion

Figure 5.1: Examples of data augmentation: (Top) Horizontal/Vertical shift; (Bottom) Flip, Occlusion, and Flip + Occlusion.

5.5 Training Procedures

Another aspect that merited exploration was the impact of domain knowledge on the classification process. To analyze this, models were replicated with incrementally higher percentages of frozen layers, meaning layers whose weights would remained unchanged throughout the training or validation process. For this, in addition to the baseline model with no frozen layers, iterations with 25%, 50% and 75% of the initial layers frozen, as well as an extreme case with all but the last layer frozen, was trained.

Regarding the remaining hyperparameters, they were initially chosen pragmatically, valuing consistency over systematic optimization. For ResNet18 models, every variation ran for a maximum of 400 epochs, and the DeiT model ran for a maximum of 50 epochs. Validation was performed after every epoch. In the case of the DeiT, early stopping was initially employed as a measure against overfitting, with a patience of 10 epochs. However, this feature did not permit us to have a full view of the model performance. The overfitting present was severe enough that the early stopping would trigger almost immediately, within the first 10 epochs. As such, in order to have a better understanding of the model learning curve, it was removed in the main experiments.

This did not negatively effect the testing portion of the experiments, as the model chosen for those was the one that maximized accuracy and minimized loss within the run, regardless if it was the result of the last epoch or not.

For batch sizes, the ResNet variant models both used a batch size of 128, and the DeiT a batch size of 32, according to GPU memory availability. ResNet models had a starting learning rate of 1×10^{-4}, while the DeiT had one of 1×10^{-5}. Both relied on the Adam optimizer, and training losses were calculated through Sparse Categorical Cross-Entropy.

While these values were chosen somewhat arbitrarily, the same setup was maintained across all experiments, ensuring that comparisons between models and datasets were fair.

5.6 Experimental Protocols

All experiments followed a consistent protocol to help comparability. Datasets were split on a subject basis, as outlined in Section 5.2, with no overlap between training, validation, and test subjects.

The ResNet variants were tested with both with no specific initialization and with convolutional weights pretrained on FER2013+. The DeiT was always used with the pre-training from ImageNet. Each model was trained a total of 20 times per initialization per dataset, accounting for all variations and combinations of freeze percentages and data augmentation methods, resulting in a total of 200 runs.

To ensure reproducibility, random seeds were fixed to 1234 both in TensorFlow and PyTorch, minimizing variability. The goal of these choices was not necessarily to maximize performance, but to ensure that the observed results directly reflect differences between model designs and dataset properties rather than experimental inconsistencies or differences.

Chapter 6

Results and Discussion

In this chapter, the results from the experiments described in Chapter 5 are presented and discussed. Accuracy was the main chosen metric to evaluate and compare the performances, used to ease the direct comparison between models. It is complemented with confusion matrices, loss progressions plots, ROC curves and genuine and impostor score distributions for a more thorough examination. Due to the amount of graphs and to avoid too many redundancies, while they will be discussed, only representative examples will be shown.

6.1 Experiments Description

Two alternative formulations were considered for the pain recognition task, binary classification and multiclass classification. Binary classification aims to just determine whether pain is present or not, regardless of its intensity, whereas multiclass classification would allow for distinguishing different levels of pain. Although the latter could provide a more nuanced understanding, for the purposes of this dissertation, only the binary classification approach was explored.

As per the specifications of Chapter 5, the performance of each model was evaluated in each of the datasets when used, in conjunction with basic shifting augmentations (that remained present in later groups), horizontal flips, random occlusions, and a combination of horizontal flips and random occlusions. Additionally, these were repeated for freeze percentages of 25%, 50%, 75% and "all but the last layer" identified as 100%. The ResNet18 models were evaluated both with random initialization and with pre-trained weights from FER2013+, and the DeiT was always used with pre-trained weights from Imagenet.

The experiments were structured according to the models introduced in Chapter 4, but here briefly detailed:

- **Baseline ResNet18** – An altered version of ResNet18. This served as a baseline to assess whether a model known to perform well on multiclass emotion recognition could transfer effectively to pain detection.
- **ResNet18 with FAU-based MLP** – To enrich the baseline approach, a parallel multi-layer perceptron (MLP) was introduced to process Facial Action Units (FAUs).

- **Data-efficient image Transformer (DeiT) model** – Finally, a transformer architecture, previously shown to be effective in pain detection tasks.

With this experimental setup, the following sections presents the results, comparing model performance across all datasets and highlighting the effects of feature integration and model complexity.

6.2 Overview of Experimental Results

The following Table 6.1 chronicles the summation of the models overall performance. Mean accuracies were calculated taking into account all experiments. Due to the specific nature of the variations made, these aggregated values do not reflect the nuances and differences of the different freeze and augmentation setups, and this table is therefore a crude comparison between models.

Table 6.1: Test accuracy (Mean ± Standard Deviation) across datasets.

Model	**BioVid**	**MIntPAIN**
ResNet18 (baseline)	51.92 ± 1.86	49.13 ± 1.28
ResNet18 (FER2013+ pretrained)	51.13 ± 1.80	49.35 ± 1.48
ResNet18 + AU (baseline)	51.91 ± 1.84	50.43 ± 1.19
ResNet18 + AU (FER2013+ pretrained)	52.06 ± 1.64	50.66 ± 1.69
DeiT (ImageNet pretrained)	54.04 ± 2.01	51.17 ± 2.11

For binary classification, and with equal representation of both classes of all sets used, 50% is the baseline for random choice. Given that the mean values do not deviate much from that value; it can be surmised that the learning task was complex, and the accuracies reached reflect that difficult climb. Nonetheless, conclusions can still be drawn.

All models and initializations had inferior accuracies on MIntPAIN, notably with the ResNet18 model having a mean below 50%, and therefore below random chance.

Overall, a clear increase in performance is seen with the increase in model complexity and input size, speaking to the strengths of more recent approaches. Alternatively, the difference is not too notorious, further emphasizing the difficulty of the task.

Getting a better look at each model in particular will further distinguish these in their respective strengths and weaknesses.

6.3 ResNet18

This section details the results of the experiments made on the simplest of our approaches, the ResNet18 model.

Before discussing the experiments on the pain based datasets, a run on the FER2013+ dataset was made. Both to generate weights to be later used in initialization for the ResNet18 models, and to also establish a comparison to emotion recognition. The final accuracy, without frozen layers and having both random occlusion and horizontal flip, was of 76.71%. This value, comparing with

the ones of Table 6.1, is extremely good, even if mid-range for state of the art standards of emotion recognition. The weights from this run were the ones used for initialization.

Moving on to the experiments, Tables 6.2 - 6.5 detail all experimental accuracies from runs made with the model.

Table 6.2: ResNet18 (baseline) accuracies on the BioVid dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	53.5	52.6	51.6	52.1	50.5	52.06	3.0
HorFlip	53.8	53.8	51.2	51.6	50.3	52.14	3.5
Occlusion	56.0	55.2	50.4	49.6	50.0	52.24	6.4
HorFlip+Occ	53.6	50.0	50.3	50.3	51.9	51.24	3.6
Mean	54.23	52.90	50.88	50.90	50.70		
Standard Deviation	1.19	2.21	0.63	1.15	0.89		

Table 6.3: ResNet18 (FER2013+) accuracies on the BioVid dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	52.5	52.4	51.5	51.5	48.5	51.28	4.0
HorFlip	53.9	55.0	51.6	51.9	49.2	52.32	5.8
Occlusion	53.2	53.2	49.5	49.4	50.0	51.06	3.8
HorFlip+Occ	50.0	50.3	49.9	49.4	49.8	49.88	0.9
Mean	52.40	52.73	50.63	50.55	49.38		
Standard Deviation	1.70	1.95	1.08	1.34	0.68		

Table 6.4: ResNet18 (baseline) accuracies on the MIntPAIN dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	50.7	48.9	50.0	49.7	50.4	49.94	1.8
HorFlip	49.4	46.4	50.0	48.3	50.8	48.98	4.4
Occlusion	47.8	50.6	46.3	48.7	49.6	48.60	4.3
HorFlip+Occ	48.6	49.7	48.0	49.0	49.7	49.00	1.7
Mean	49.13	48.90	48.58	48.93	50.13		
Standard Deviation	1.24	1.81	1.79	0.59	0.57		

Table 6.5: ResNet18 (FER2013+) accuracies on the MIntPAIN dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	48.4	47.4	49.8	48.9	48.9	48.68	2.4
HorFlip	47.3	50.5	49.8	47.3	48.9	48.76	3.2
Occlusion	50.2	52.2	52.9	49.3	48.9	50.70	4.0
HorFlip+Occ	49.1	49.1	51.1	48.1	49.0	49.28	3.0
Mean	48.75	49.80	50.90	48.40	48.93		
Standard Deviation	1.22	2.04	1.47	0.89	0.05		

Figures 6.1 - 6.4 were the result from the model run on the BioVid dataset using no specific initialization, no frozen layers, and random occlusion as augmentation.

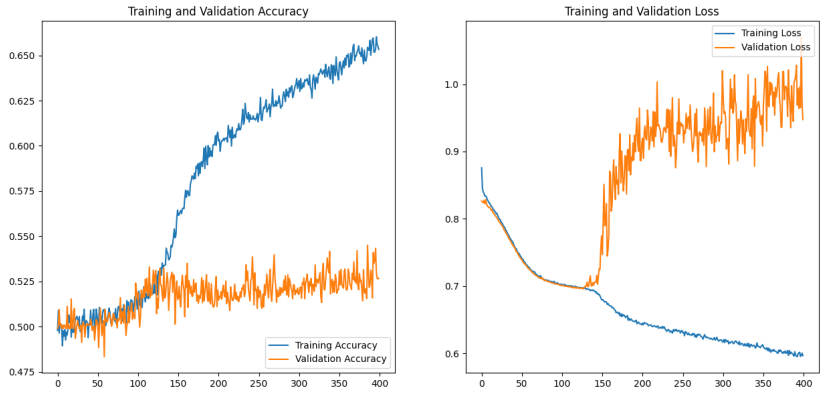


Figure 6.1: Accuracy and Loss progression for training and validation sets using the ResNet18 model.

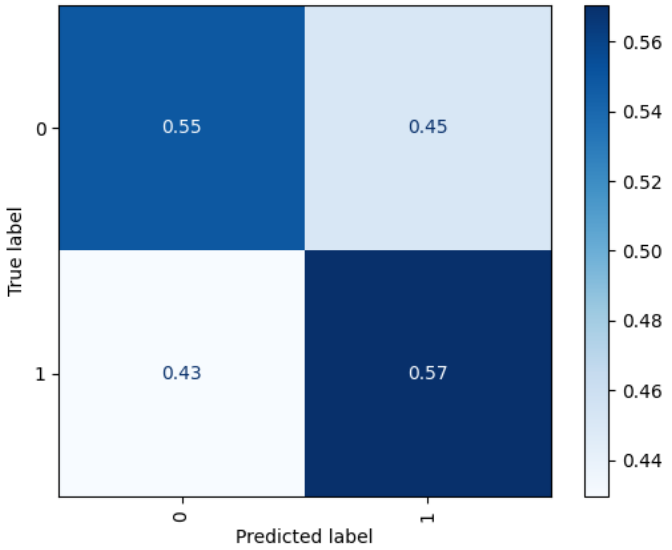


Figure 6.2: Confusion Matrix for the ResNet18 model (0 - No Pain, 1 - Pain).

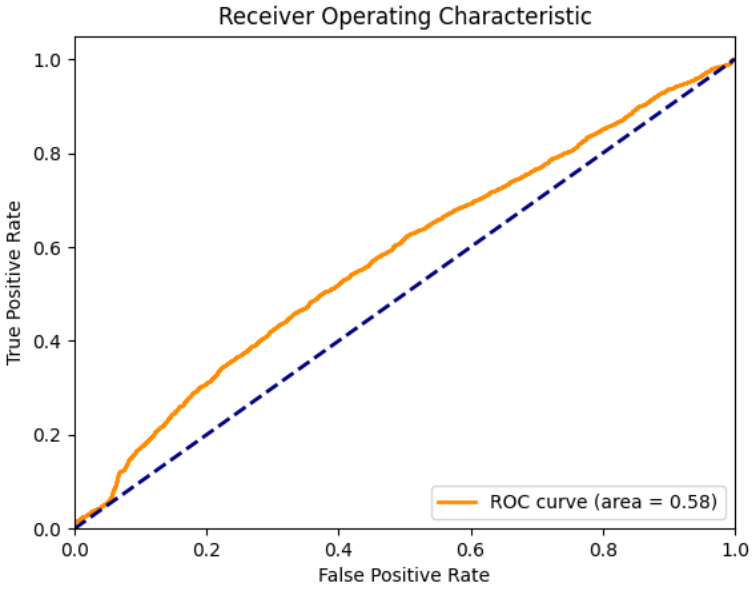


Figure 6.3: ROC Curve for the ResNet18 model.

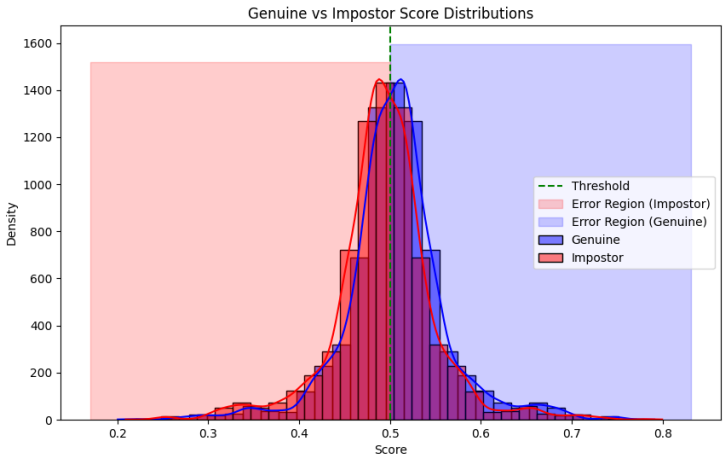


Figure 6.4: Genuine vs Impostor score distributions for the ResNet18 model.

6.4 ResNet18 + MLP

This section details the results of the experiments made on the ResNet18 added with an MLP for FAU parallel processing. Tables 6.6 - 6.9 detail all experimental accuracies from runs made with the model. While there were no pretrained weights for the fully connected branch, initialization with the FER2013+ weights for the convolutional branch was performed.

Table 6.6: ResNet18 + MLP(baseline) accuracies on the BioVid dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	54.2	53.4	52.5	53.0	51.3	52.88	2.9
HorFlip	54.5	54.2	52.0	52.1	50.9	52.74	3.6
Occlusion	57.0	56.5	52.0	51.3	51.2	53.60	5.8
HorFlip+Occ	54.6	51.2	51.5	51.4	53.0	52.34	3.4
Mean	55.20	53.83	52.00	51.95	51.60		
Standard Deviation	1.29	2.19	0.41	0.79	0.95		

Table 6.7: ResNet18 + MLP(FER2013+) accuracies on the BioVid dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	53.2	53.0	52.2	52.3	49.7	52.08	3.5
HorFlip	54.3	55.5	52.3	52.5	50.1	52.94	5.4
Occlusion	54.8	54.0	50.8	50.5	51.2	52.26	4.3
HorFlip+Occ	51.5	51.2	50.9	50.5	50.8	50.98	1.0
Mean	53.45	53.43	51.55	51.45	50.45		
Standard Deviation	1.46	1.80	0.81	1.10	0.68		

Table 6.8: ResNet18 + MLP (baseline) accuracies on the MIntPAIN dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	51.2	49.4	50.5	50.2	50.9	50.44	1.8
HorFlip	50.2	47.2	50.8	49.1	51.6	49.78	4.4
Occlusion	50.0	52.6	48.8	50.7	51.4	50.70	3.8
HorFlip+Occ	50.4	51.5	49.8	50.8	51.5	50.80	1.7
Mean	50.45	50.18	49.98	50.20	51.35		
Standard Deviation	0.53	2.39	0.89	0.78	0.31		

Table 6.9: ResNet18 + MLP (FER2013+) accuracies on the MIntPAIN dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	49.2	48.2	50.6	49.7	49.7	49.48	2.4
HorFlip	48.6	51.8	50.8	48.3	49.9	49.88	3.5
Occlusion	52.2	54.0	54.4	50.9	50.5	52.40	3.9
HorFlip+Occ	50.7	50.7	52.7	49.7	50.6	50.88	3.0
Mean	50.18	51.18	52.13	49.65	50.18		
Standard Deviation	1.61	2.41	1.79	1.06	0.44		

Figures 6.5 - 6.8 were the result from the model run on the BioVid dataset, using the FER2013+ initialization weights, with all layers frozen but one, and horizontal flip as augmentation.

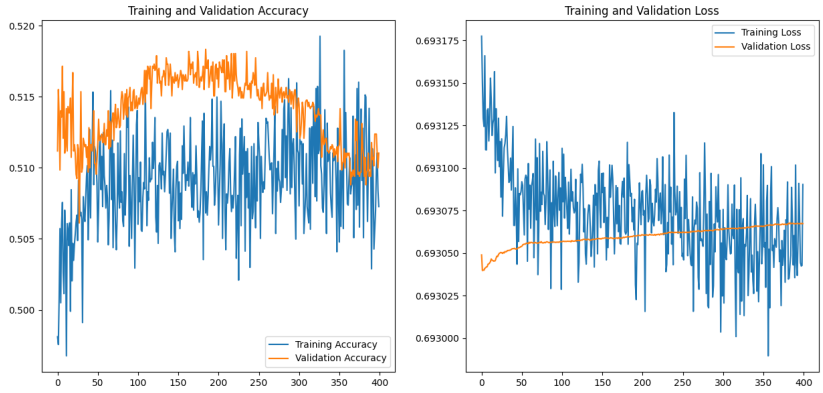


Figure 6.5: Accuracy and Loss progression for training and validation sets using the MLP model.

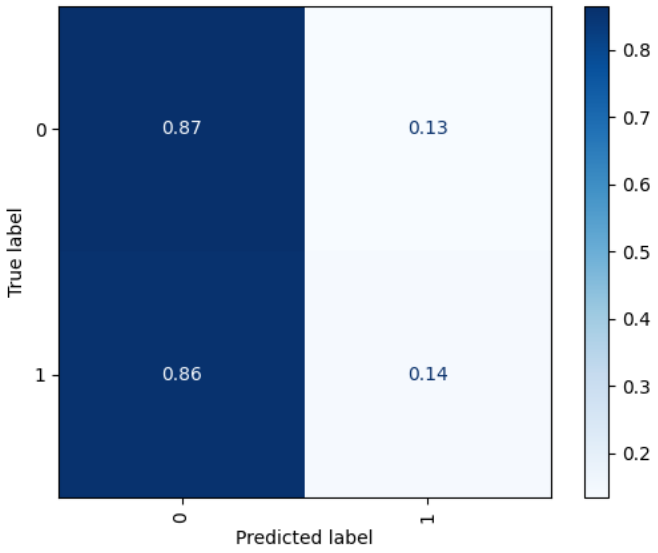


Figure 6.6: Confusion Matrix for the MLP model (0 - No Pain, 1 - Pain).

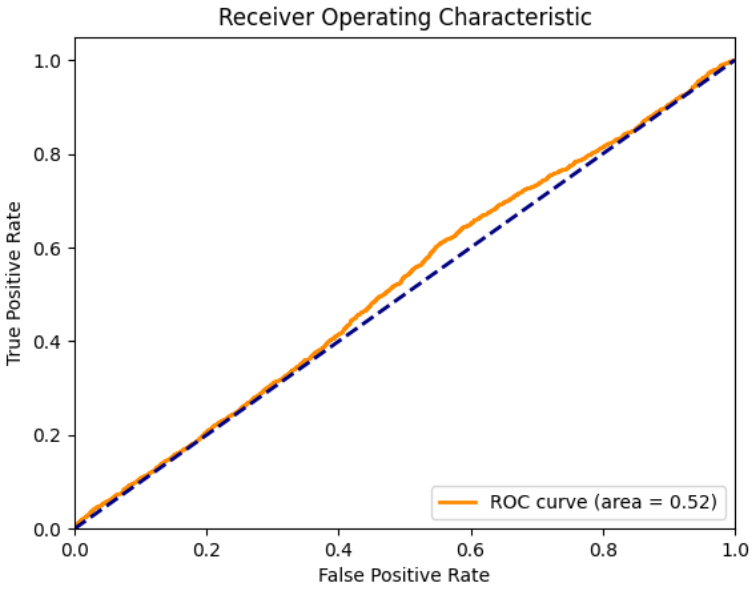


Figure 6.7: ROC Curve for the MLP model.

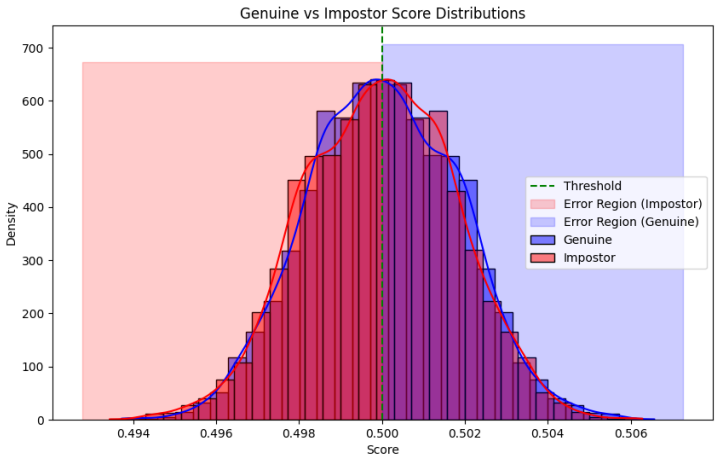


Figure 6.8: Genuine vs Impostor score distributions for the MLP model.

6.5 DeiT

Finally, this section details the results of the experiments made on the Data-efficient image Transformer (DeiT). Tables 6.10 and 6.11 detail all experimental accuracies from runs made with the model. In this case, the model used always included pretraining on the ImageNet database.

Table 6.10: DeiT accuracies on the BioVid dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	54.2	54.3	53.9	53.1	50.6	53.22	3.7
HorFlip	54.4	54.6	56.0	53.8	50.7	53.90	5.3
Occlusion	57.1	56.5	54.6	54.7	50.7	54.72	6.4
HorFlip+Occ	54.6	55.3	56.9	54.2	50.7	54.34	6.2
Mean	55.08	55.18	55.35	53.95	50.68		
Standard Deviation	1.36	0.98	1.35	0.68	0.05		

Table 6.11: DeiT accuracies on the MIntPAIN dataset with different augmentations and freeze percentages.

Augmentations	**0%**	**25%**	**50%**	**75%**	**100%**	**Mean**	**Range**
Simple	51.4	50.5	52.2	50.7	50.5	51.06	1.7
HorFlip	50.0	47.1	54.7	50.4	51.2	50.68	7.6
Occlusion	48.7	51.8	50.2	53.7	50.3	50.94	5.0
HorFlip+Occ	49.5	55.0	54.3	52.8	48.5	52.02	6.5
Mean	49.90	51.10	52.85	51.90	50.13		
Standard Deviation	1.13	3.27	2.08	1.61	1.15		

Figures 6.9 - 6.12 were the result from the model run on the MIntPAIN dataset, with the Imagenet initialization, with 50% freeze percentage, and both random occlusion and horizontal flip as augmentations.

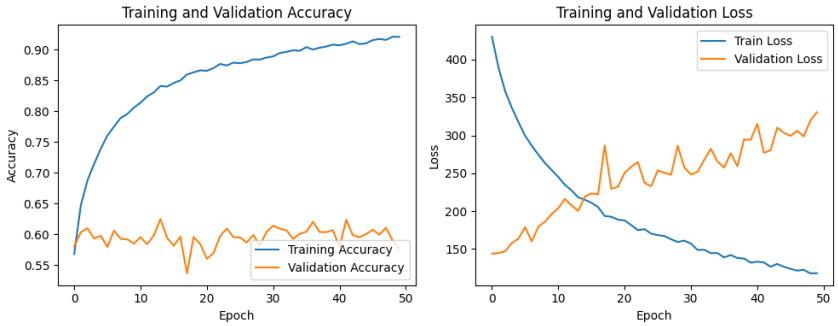


Figure 6.9: Accuracy and Loss progression for training and validation sets using the DeiT model.

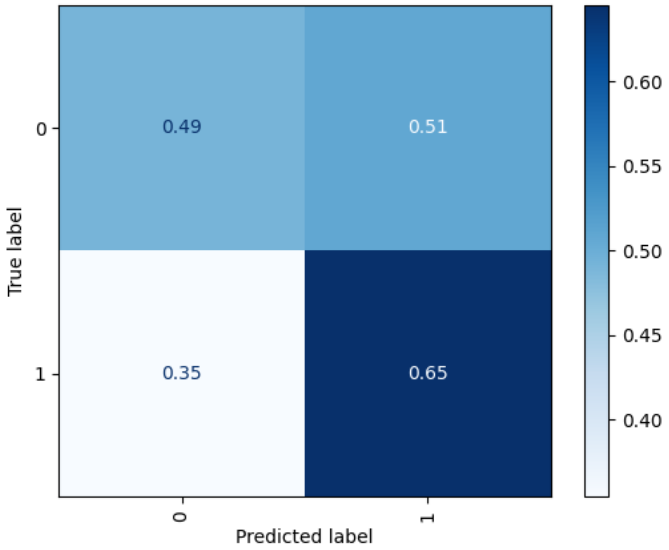


Figure 6.10: Confusion Matrix for the DeiT model (0 - No Pain, 1 - Pain).

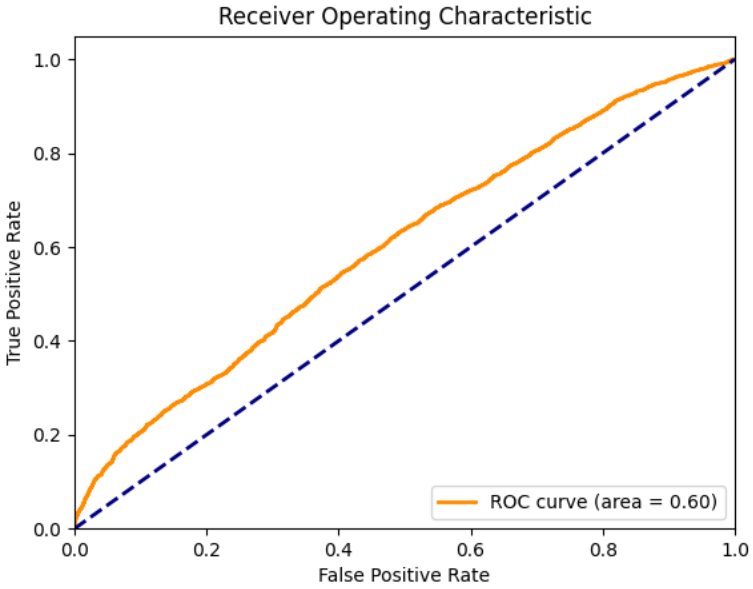


Figure 6.11: ROC Curve for the DeiT model.

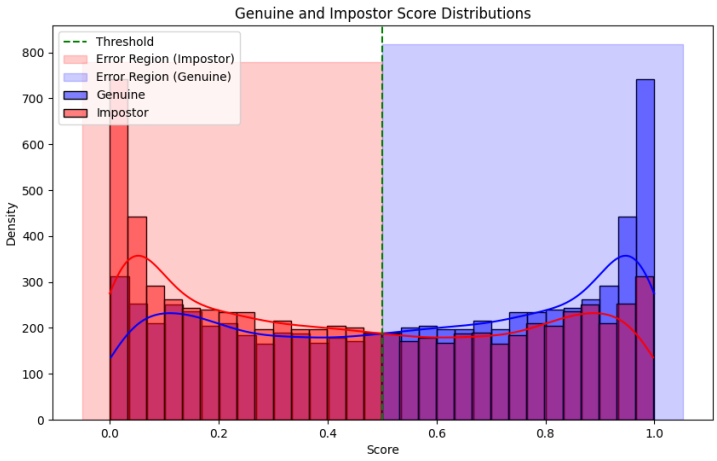


Figure 6.12: Genuine vs Impostor score distributions for the DeiT model.

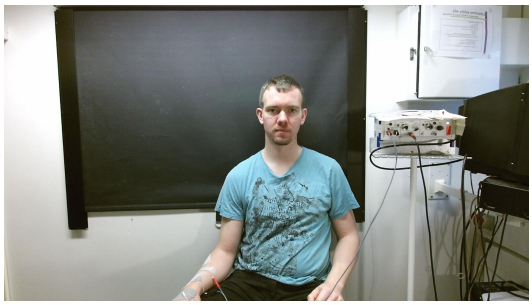
6.6 Comparative Analysis

This section synthesizes results to highlight comparative insights and establish patterns detected.

6.6.1 Dataset-specific differences

The results obtained are subpar when compared to the literature. While prior work in binary classification often reported accuracies above 70% (see Chapter 3), our best configuration reached 57.1%. This gap highlights the difficulty of transferring models trained on generic emotion datasets to the subtler task of pain recognition.

Nonetheless, parallels are still present to be analysed, and conclusions can be made from them. As could already be ascertained from Table 6.1, MIntPAIN saw lower results than BioVid yielding the lowest observed accuracy, 46.3% using ResNet18, with no initialization, 25% freeze and only horizontal flip. Some distinctions between the datasets can be highlighted as potentially creating this difference. BioVid contained almost three times as many samples and four times as many subjects, with more controlled capture conditions. As can be seen by the comparison in Figure 6.13, data for the BioVid database was captured under more focused conditions, with the camera closer to the face, and more consistent lighting. These factors likely explain why BioVid consistently outperformed MIntPAIN, regardless of model.



(a) Sub1 example frame from MIntPAIN dataset.

(b) 072514 example frame from BioVid dataset.

Figure 6.13: Dataset comparison.

6.6.2 Architecture-specific differences

Regarding inter model differences, the hierarchy of performances is as expected, with the base ResNet18 performing the worst and the DeiT the best. However, the difference between the ResNet18 with access to FAU information and the DeiT was not that pronounced, shedding light on the effectiveness of FAUs in pain classification. This inclusion globally improved scores across the board for the ResNet18 performance, effectively serving as an improved version of the base model. The FAUs were not part of the original dataset labels, but instead automatically detected. Had the datasets already included this type of detailed and reviewed information, one can extrapolate that the improvement would have been even greater, with proper annotations aiding the classification.

For the ResNet18 models, the differing initializations did not lead to a uniform improvement or deterioration of results. If we go by the overall values in Table 6.1, the FER2013+ led to lower performance in the BioVid dataset, but slightly higher performance on the MIntPAIN dataset. This in itself is an interesting insight, suggesting that domain knowledge over emotion classification does not translate cleanly to pain classification, and should be judged in a case by case basis.

Training times were also variable, although dependent on more than just model architecture differences. Dataset (more accurately, size difference between them), batch size, number of epochs, freeze percentage and augmentations were the main contributors. Generally, BioVid based training took longer than MIntPAIN based training, using multiple data augmentation methods universally increased training times, and using higher freeze percentages decreased them. While models were always trained with different batch sizes and for a differing number of epochs (ResNet18 models were always trained for at maximum 400 epochs and using a batch size of 128, while the DeiT model was trained for at maximum 50 epochs with a batch size of 32), it is safe to say that DeiT took more computational power and would've taken more time to train if parameters were equal. The ResNet18 models took between two and nine hours, corresponding to the extremes of simple augmentation and maximum freeze, and both studied augmentation methods with no frozen layers, while throughout all runs the DeiT model remained at the domain of one and a half to three hours.

6.6.3 Effect of augmentations and freeze levels

Regarding the different augmentation methods, analysis will focus on the lower freeze percentages, as these allow for more variation based on the differing augmentations. In the ResNet18 models, random occlusion alone resulted in the highest scores, with the addition of 50% chance of horizontal flipping effectively reversing that improvement. This suggests that, at least for the ReNet18 models, introducing changes too complex can actively harm the models ability to learn, and that horizontally flipping the images, mirroring facial expressions across the vertical axis, can distort pain-related features in a way that confuses the model. The same can be said for the DeiT used in the BioVid dataset, but when used in the MIntPAIN dataset, the combination of occlusion and horizontal flipping yielded the best results. The drawn conclusion is that DeiT is more flexible than the ResNet18 alternatives, and easily adapts to multiple image variations.

Finally, freeze percentages also had a clear effect on accuracy. At the maximum 100%, or all but the final layer frozen, models behaved very closely to 50% across augmentations, demonstrated in Figure 6.5. This marked a convergence towards random choice, which is to be expected since, as previously stated, both labels are equally represented in the evaluation set. In most cases, it is a decrease in performance. This aligns with the principle of backpropagation, since frozen layers prevent weight updates and limit learning capacity. It is also visible in the training/validation accuracy progression curves, where the peak accuracy achieved within the training dataset lowering with increasing freeze percentage.

6.6.4 Training dynamics and overfitting

The progression of training and validation curves further illustrates the models' limitations. At low freeze percentages, validation loss diverged almost immediately while training loss decreased steadily. This is visible cleanly in Figure 6.1, where the validation loss accompanied the training loss at the start, before diverging explosively. While this can be due to multiple reasons, the most likely one is overfitting.

The DeiT model showed the strongest signs of this behavior, where the training dataset reaches above 80% training accuracy in just a few epochs, as seen in Figure 6.9, with little to no progression for the validation accuracy. The ResNet models overfitted more slowly, only reaching about 60% training accuracy before plateauing. In some cases, validation accuracy improved slightly in the early epochs, but then stagnated while the training accuracy is still rising. These dynamics suggest that both model families struggled to extract stable and generalizable patterns from the available data, regardless of augmentations.

6.6.5 Graphical evaluation of classifier behavior

To complement the numerical metrics, representative examples of ROC curves, confusion matrices, and genuine vs impostor distributions were provided above. These were selected to illustrate different architectures and configurations, rather than to exhaustively cover all runs. While not

comprehensive of all results, these examples are consistent with the general tendencies observed across trials.

The ROC curves (Figs. 6.3, 6.7, 6.11) demonstrate the limited discriminative ability of the models. In all three cases, the curves remain relatively close to the diagonal, indicating performance near chance level. The slight elevation of the curve above the diagonal in Figures 6.3 and 6.11 reflects the higher accuracy these particular models had over the other examples.

The confusion matrices (Figs. 6.2, 6.6, 6.10) provide further evidence of limitations. Predictions are distributed across both classes, but with no strong dominance of correct classifications, as can be seen in the closeness of coefficients of Figures 6.2 and 6.10. Only in some cases, where the models display an extreme bias towards one class, do we usually see a stark difference in scales between the predictions. This is illustrated in Figure 6.5, and is how most models behaved when they were closer and closer to 50% accuracy, essentially not detecting any distinguishing features for one way of the other, and classifying almost all, if not all, images as the first class “No Pain”, or second class "Pain".

Finally, the genuine versus impostor score distributions (Figs. 6.4, 6.8, 6.12) highlight the lack of separation in the representations, even in the "better" performing models. The heavy overlap between distributions indicates that the models were unable to form embeddings that could reliably distinguish between pain and no-pain instances, showing that both model families struggled to generalize effectively. Figure 6.12 shows promise, with some distancing of the peaks from the center, but the heavy overlap is still maintained.

6.7 Difficulties and Limitations

The limitations observed in this study seem to stem largely from dataset and preprocessing constraints. Differences in dataset size and acquisition methods, and the absence of peer-reviewed facial landmark or FACS-based annotations, meaning that other than intensity levels, there were no other potentially useful features for facial emotion detection detailed in the datasets, likely contributed to the difficulty in adapting to the validation and testing groups. Despite careful partitioning, equal class representation, and prevention of contamination in the partitioned sets, the models struggled to generalize. This suggests that they were unable to extract sufficient information to discriminate data effectively.

One important factor is the restriction to grayscale images. This choice may have removed valuable cues such as color-based changes in skin tone, which may carry information about pain. Supporting this, prior work with DeiT on RGB BioVid inputs reported stronger results. [56]

Another limitation is image compression. To allow experimentation with initialization across architectures, inputs were resized to relatively low resolutions. While this yielded acceptable results in the simpler FER2013+ emotion recognition task, pain recognition requires detecting subtler facial cues. Excessive compression likely suppressed these details, further enhancing the difficulty of the task.

These findings suggest that the chosen modality (grayscale only) and preprocessing choices (heavy resizing) interacted with the pre-existing dataset constraints to hinder model performance. Future work should explore RGB input representations, higher-resolution training, and domain-specific augmentations and annotations, which may better perform the more nuanced task of pain detection.

Chapter 7

Conclusions and Future Work

This chapter bookends the dissertation, with final conclusions and remarks for future work.

7.1 Conclusions

This work has provided a study on automated pain recognition using only facial images, exploring multiple architectures and variations on training strategies. Experiments focused on binary pain classification using a convolutional model (ResNet18) and a transformer-based approach (DeiT), with extra evaluation of Facial Action Units (FAUs) as an auxiliary feature. Overall performance remained below the state-of-the-art, highlighting the challenges in recognizing pain compared to generic emotion recognition. Table 7.1 presents the best obtained scores in either dataset for each model.

Table 7.1: Review of the best accuracy scores, in percentage (%), per dataset, per model.

Model	**BioVid**	**MIntPAIN**
ResNet18	56.0	52.9
ResNet18 + MLP	57.0	54.4
DeiT	57.1	55.0

It was found that dataset characteristics strongly influenced results. The models consistently performed better in BioVid, likely due to its larger size, higher subject diversity, and more controlled capture conditions. MIntPAIN, with fewer samples and more variable conditions, produced overall lower accuracies. Regarding model architectures, the baseline ResNet18 performed worst, with FAU integration improving results across the board, demonstrating the value of incorporating more domain-specific features. The DeiT transformer achieved the highest accuracies overall, particularly on BioVid, calling to the advantages of modern attention-based architectures. Pretraining of the ResNet18 models on FER2013+ did not lead to a consistent improvement in performance, suggesting limited knowledge transfer from generic emotion recognition to pain detection.

Data augmentation and freeze percentages also significantly affected outcomes. Random occlusion often improved ResNet18 performance, while horizontal flips sometimes reduced the resulting accuracy, suggesting a level of asymmetry in pain expressions. High freeze percentages universally caused a convergence into random chance, and training dynamics indicated early overfitting, particularly for DeiT.

The limitations identified contextualize these findings. The lack of peer-reviewed facial landmark or FACS-based annotations within the database, and other self imposed constraints, such as grayscale-only inputs and heavy resizing for model compatibility, likely hindered generalization. Combined with the inherent dataset limitations, models may simply have lacked sufficient information to learn discriminative pain cues. These results underscore the difficulty of automated pain recognition, and the importance of carefully aligning data quality, feature selection, and model design.

7.2 Finishing Remarks

Despite the modest model performance achieved, this study provides meaningful insights into the strengths and limitations of current approaches to pain recognition. It demonstrates that the inclusion of domain-specific features can improve performance, but generalization remains a major challenge due to the subtle and variable nature of pain expression. Importantly, this work establishes a clear comparative framework used for evaluating different architectures and training strategies, highlighting the interplay between datasets, augmentations and pretraining.

The results from this framework reinforce that successful pain recognition requires high-quality data and architectures capable of capturing subtler features than those present in emotional recognition. By identifying the key circumstances that limited it, this work can serve as a foundation for future studies to build upon.

7.3 Future Work

Future research should focus on creating larger and more heavily annotated datasets specifically tailored to pain recognition. This would help mitigate overfitting and improve generalization. Incorporating multimodal signals, such as the physiological data present in ECGs, EMGs, or EEGs, alongside facial expressions, may enhance robustness, though at the cost of ease of access and generalization to other datasets. Hybrid architectures combining convolutional backbones with attention mechanisms, or leveraging temporal dynamics, could better capture subtle cues.

Further work could explore more effective integration of FAUs or other domain-specific features, ideally with verified annotations. Self-supervised learning approaches could also potentially make better use of limited labeled data. Combining transformer architectures with FACS-oriented inputs also represents a promising direction.

Overall, these directions provide a roadmap for developing more accurate and generalizable automated pain recognition systems, helping to move beyond proof-of-concept and into real world applications.

Bibliography

[1] C Darwin. *The Expression of the Emotions in Man and Animals*. John Murray, 1872.

[2] Gadi Gilam et al. "What Is the Relationship between Pain and Emotion? Bridging Constructs and Communities". In: *Neuron* 107 (1 July 2020), p. 17. ISSN: 10974199. DOI: 10.1016/J.NEURON.2020.05.024. URL: /pmc/articles/PMC7578761//pmc/articles/PMC7578761/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC7578761/.

[3] Emad Kasaeyan Naeini et al. "Pain Recognition With Electrocardiographic Features in Postoperative Patients: Method Validation Study". In: *Journal of Medical Internet Research* 23 (5 May 2021), e25079. ISSN: 1438-8871. DOI: 10.2196/25079.

[4] Guozhen Zhao et al. "Multi-Target Positive Emotion Recognition From EEG Signals". In: *IEEE Transactions on Affective Computing* 14 (1 Jan. 2023), pp. 370–381. ISSN: 1949-3045. DOI: 10.1109/TAFFC.2020.3043135. URL: https://ieeexplore.ieee.org/document/9286502/.

[5] Nasser Ali Malik. "Revised definition of pain by 'International Association for the Study of Pain': Concepts, challenges and compromises". In: *Anaesthesia, Pain & Intensive Care* 24 (5 Oct. 2020), pp. 481 –483. ISSN: 2220-5799. DOI: 10.35975/apic.v24i5.1352. URL: http://apicareonline.com/index.php/APIC/article/view/1352.

[6] John D. Loeser and Rolf-Detlef Treede. "The Kyoto protocol of IASP Basic Pain Terminology ". In: *Pain* 137 (3 July 2008), pp. 473–477. ISSN: 0304-3959. DOI: 10.1016/j.pain.2008.04.025. URL: https://journals.lww.com/00006396-200807310-00005.

[7] Dominik Mischkowski et al. "Pain or nociception? Subjective experience mediates the effects of acute noxious heat on autonomic responses". In: *Pain* 159 (4 Apr. 2018), pp. 699–711. ISSN: 0304-3959. DOI: 10.1097/j.pain.0000000000001132. URL: https://journals.lww.com/00006396-201804000-00010.

[8] Asimina Lazaridou et al. "Pain Assessment". In: Elsevier, Jan. 2018, 39–46.e1. DOI: 10.1016/B978-0-323-40196-8.00005-X. URL: https://linkinghub.elsevier.com/retrieve/pii/B978032340196800005X.

[9] *IASP Terminology*. URL: https://www.iasp-pain.org/resources/terminology/.

[10] Troels S. Jensen et al. “A new definition of neuropathic pain”. In: *Pain* 152 (10 Oct. 2011), pp. 2204–2205. ISSN: 0304-3959. DOI: 10.1016/j.pain.2011.06.017.

[11] Eva Kosek et al. “Do we need a third mechanistic descriptor for chronic pain states?” In: *Pain* 157 (7 July 2016), pp. 1382–1386. ISSN: 0304-3959. DOI: 10.1097/j.pain.0000000000000507.

[12] Barry Eliot Cole. “Pain Management: Classifying, Understanding, and Treating Pain”. In: 2003, pp. 23–30.

[13] John D Loeser and Ronald Melzack. “Pain: an overview”. In: *The Lancet* 353.9164 (1999), pp. 1607–1609. ISSN: 0140-6736. DOI: https://doi.org/10.1016/S0140-6736(99)01311-2. URL: https://www.sciencedirect.com/science/article/pii/S0140673699013112.

[14] Rolf-Detlef Treede et al. “A classification of chronic pain for ICD-11”. In: *Pain* 156 (6 June 2015), pp. 1003–1007. ISSN: 0304-3959. DOI: 10.1097/j.pain.0000000000000160. URL: https://journals.lww.com/00006396-201506000-00006.

[15] Rolf-Detlef Treede et al. “Chronic pain as a symptom or a disease: the IASP Classification of Chronic Pain for the International Classification of Diseases (ICD-11)”. In: *Pain* 160 (1 Jan. 2019), pp. 19–27. ISSN: 0304-3959. DOI: 10.1097/j.pain.0000000000001384. URL: https://journals.lww.com/00006396-201901000-00003.

[16] Joachim Scholz et al. “The IASP classification of chronic pain for ICD-11: chronic neuropathic pain”. In: *Pain* 160 (1 Jan. 2019), pp. 53–59. ISSN: 0304-3959. DOI: 10.1097/j.pain.0000000000001365. URL: https://journals.lww.com/00006396-201901000-00007.

[17] Paul Ekman and Wallace V. Friesen. “Constants across cultures in the face and emotion.” In: *Journal of Personality and Social Psychology* 17 (2 1971), pp. 124–129. ISSN: 1939-1315. DOI: 10.1037/h0030377.

[18] Paul Ekman and Wallace V. Friesen. *Facial Action Coding System*. Jan. 1978. DOI: 10.1037/t27734-000. URL: https://doi.apa.org/doi/10.1037/t27734-000.

[19] Paul Ekman and Erika L. Rosenberg. *What the Face RevealsBasic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)*. Oxford University Press, Apr. 2005. ISBN: 9780195179644. DOI: 10.1093/acprof:oso/9780195179644.001.0001.

[20] American Psychological Association. *Emotion*. [Accessed: April. 20, 2025]. URL: https://dictionary.apa.org/emotion.

[21] Paul Ekman. “Universals and cultural differences in facial expressions of emotion.” In: *Nebraska Symposium on Motivation* 19 (1971), pp. 207–283. ISSN: 0146-7875(Print).

[22] ADAM KENDON and STUART J. SIGMAN. "Commemorative essay. Ray L. Birdwhistell (1918-1994)". In: *Semiotica* 112 (3-4 1996). ISSN: 0037-1998. DOI: 10.1515/semi.1996.112.3-4.231.

[23] Ruicong Zhi, Mengyi Liu, and Dezheng Zhang. "A comprehensive survey on automatic facial action unit analysis". In: *The Visual Computer* 36 (5 May 2020), pp. 1067–1093. ISSN: 0178-2789. DOI: 10.1007/s00371-019-01707-5.

[24] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *Nature 2015 521:7553* 521 (7553 May 2015), pp. 436–444. ISSN: 1476-4687. DOI: 10.1038/nature14539. URL: https://www.nature.com/articles/nature14539.

[25] F. Rosenblatt. "The perceptron: A probabilistic model for information storage and organization in the brain." In: *Psychological Review* 65 (6 1958), pp. 386–408. ISSN: 1939-1471. DOI: 10.1037/h0042519. URL: https://doi.apa.org/doi/10.1037/h0042519.

[26] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. *Rectifier Nonlinearities Improve Neural Network Acoustic Models*. Tech. rep. 2013.

[27] Kaiming He et al. "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification". In: (Feb. 2015). URL: http://arxiv.org/abs/1502.01852.

[28] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)". In: (Feb. 2016). URL: http://arxiv.org/abs/1511.07289.

[29] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[30] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. "Searching for Activation Functions". In: (Oct. 2017). URL: http://arxiv.org/abs/1710.05941.

[31] Diganta Misra. "Mish: A Self Regularized Non-Monotonic Activation Function". In: *ArXiv* (Aug. 2020). URL: http://arxiv.org/abs/1908.08681.

[32] Emanuele Lattanzi, Matteo Donati, and Valerio Freschi. "Exploring Artificial Neural Networks Efficiency in Tiny Wearable Devices for Human Activity Recognition". In: *Sensors* 22 (7 Mar. 2022), p. 2637. ISSN: 1424-8220. DOI: 10.3390/s22072637. URL: https://www.mdpi.com/1424-8220/22/7/2637.

[33] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings* (Sept. 2014). URL: https://arxiv.org/abs/1409.1556v6.

[34] Christian Szegedy et al. “Going deeper with convolutions”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 07-12-June-2015 (Oct. 2015), pp. 1–9. ISSN: 10636919. DOI: 10.1109/CVPR.2015.7298594.

[35] Ashish Vaswani et al. “Attention Is All You Need”. In: (Aug. 2017).

[36] Alexey Dosovitskiy et al. “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: (June 2020).

[37] Ilya Loshchilov and Frank Hutter. “Decoupled Weight Decay Regularization”. In: *International Conference on Learning Representations*. Jan. 2017. URL: https://api.semanticscholar.org/CorpusID:53592270.

[38] Kanishka Tyagi et al. “Unsupervised learning”. In: *Artificial Intelligence and Machine Learning for EDGE Computing* (Jan. 2022), pp. 33–52. DOI: 10.1016/B978-0-12-824054-0.00012-5.

[39] Longlong Jing and Yingli Tian. “Self-Supervised Visual Feature Learning With Deep Neural Networks: A Survey”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43 (11 Nov. 2021), pp. 4037–4058. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2020.2992393.

[40] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. “Distilling the Knowledge in a Neural Network”. In: *ArXiv* (Mar. 2015).

[41] Steffen Walter et al. “The biovid heat pain database data for the advancement and systematic validation of an automated pain recognition system”. In: *2013 IEEE International Conference on Cybernetics (CYBCO)*. IEEE, June 2013, pp. 128–131. ISBN: 978-1-4673-6469-0. DOI: 10.1109/CYBConf.2013.6617456. URL: http://ieeexplore.ieee.org/document/6617456/.

[42] Mohammad A. Haque et al. “Deep multimodal pain recognition: A database and comparison of spatio-temporal visual modalities”. In: *Proceedings - 13th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2018* (June 2018), pp. 250–257. DOI: 10.1109/FG.2018.00044.

[43] Patrick Lucey et al. “Painful data: The UNBC-McMaster shoulder pain expression archive database”. In: *2011 IEEE International Conference on Automatic Face and Gesture Recognition and Workshops, FG 2011* (2011), pp. 57–64. DOI: 10.1109/FG.2011.5771462.

[44] Joy O. Egede et al. “EMOPAIN Challenge 2020: Multimodal Pain Evaluation from Facial and Bodily Expressions”. In: *Proceedings - 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2020* (Jan. 2020), pp. 849–856. URL: http://arxiv.org/abs/2001.07739.

[45] Zheng Zhang et al. “Multimodal Spontaneous Emotion Corpus for Human Behavior Analysis”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 2016-December. IEEE, June 2016, pp. 3438–3446. ISBN: 978-1-4673-8851-1. DOI: 10.1109/CVPR.2016.374. URL: http://ieeexplore.ieee.org/document/7780743/.

[46] Maria Velana et al. “The SenseEmotion Database: A Multimodal Database for the Development and Systematic Validation of an Automatic Pain- and Emotion-Recognition System”. In: 2017, pp. 127–139. DOI: 10.1007/978-3-319-59259-6_11.

[47] Sascha Gruss et al. “Multi-Modal Signals for Analyzing Pain Responses to Thermal and Electrical Stimuli”. In: *Journal of Visualized Experiments* (146 Apr. 2019). ISSN: 1940-087X. DOI: 10.3791/59057. URL: https://app.jove.com/t/59057.

[48] Roberto Fernandes-Magalhaes et al. “Pain E-motion Faces Database (PEMF): Pain-related micro-clips for emotion research”. In: *Behavior Research Methods* 55 (7 Oct. 2022), pp. 3831–3844. ISSN: 1554-3528. DOI: 10.3758/s13428-022-01992-4.

[49] Raul Fernandez–Rojas et al. “The AI4Pain Grand Challenge 2024: Advancing Pain Assessment with Multimodal fNIRS and Facial Video Analysis”. In: *2024 12th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. IEEE, Sept. 2024, pp. 55–60. ISBN: 979-8-3315-1645-1. DOI: 10.1109/ACIIW63320.2024.00012.

[50] Babak Taati et al. “SynPAIN: A Synthetic Dataset of Pain and Non-Pain Facial Expressions”. In: *ArXiv* (Aug. 2025).

[51] P. Viola and M. Jones. “Rapid object detection using a boosted cascade of simple features”. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*. Vol. 1. IEEE Comput. Soc, 2001, pp. I–511–I–518. ISBN: 0-7695-1272-0. DOI: 10.1109/CVPR.2001.990517. URL: http://ieeexplore.ieee.org/document/990517/.

[52] Harihara Santosh Dadi and Gopala Krishna Mohan Pillutla. “Improved Face Recognition Rate Using HOG Features and SVM Classifier”. In: *IOSR Journal of Electronics and Communication Engineering* 11 (04 Apr. 2016), pp. 34–44. ISSN: 22788735. DOI: 10.9790/2834-1104013444. URL: http://iosrjournals.org/iosr-jece/papers/Vol.%2011%20Issue%204/Version-1/G1104013444.pdf.

[53] Ning Zhang, Junmin Luo, and Wuqi Gao. “Research on face detection technology based on MTCNN”. In: *Proceedings - 2020 International Conference on Computer Network, Electronic and Automation, ICCNEA 2020* (Sept. 2020), pp. 154–158. DOI: 10.1109/ICCNEA50255.2020.00040.

[54] Gnana Praveen R, Eric Granger, and Patrick Cardinal. “Deep DA for Ordinal Regression of Pain Intensity Estimation Using Weakly-Labeled Videos”. In: (Oct. 2020). URL: http://arxiv.org/abs/2010.15675.

[55] Ashish Semwal and Narendra D. Londhe. "S-PANET: A Shallow Convolutional Neural Network for Pain Severity Assessment in Uncontrolled Environment". In: *2021 IEEE 11th Annual Computing and Communication Workshop and Conference, CCWC 2021* (Jan. 2021), pp. 800–806. DOI: 10.1109/CCWC51732.2021.9376052.

[56] Safaa El Morabit and Atika Rivenq. "Pain Detection From Facial Expressions Based on Transformers and Distillation". In: *11th International Symposium on Signal, Image, Video and Communications, ISIVC 2022 - Conference Proceedings* (2022). DOI: 10.1109/ISIVC54825.2022.9800746.

[57] Álvaro Sabater-Gárriz et al. "Automated facial recognition system using deep learning for pain assessment in adults with cerebral palsy". In: *DIGITAL HEALTH* 10 (Jan. 2024). ISSN: 2055-2076. DOI: 10.1177/20552076241259664. URL: https://journals.sagepub.com/doi/10.1177/20552076241259664.

[58] Jiankang Deng et al. "RetinaFace: Single-stage Dense Face Localisation in the Wild". In: (May 2019). URL: http://arxiv.org/abs/1905.00641.

[59] Tsung-Yi Lin et al. "Feature Pyramid Networks for Object Detection". In: (Apr. 2017). URL: http://arxiv.org/abs/1612.03144.

[60] Md Maruf Monwar and Siamak Rezaei. "Pain recognition using artificial neural network". In: *Sixth IEEE International Symposium on Signal Processing and Information Technology, ISSPIT* (2006), pp. 28–33. DOI: 10.1109/ISSPIT.2006.270764.

[61] Sheryl Brahnam et al. "SVM classification of neonatal facial images of pain". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 3849 LNAI (2006), pp. 121–128. ISSN: 03029743. DOI: 10.1007/11676935_15. URL: https://www.researchgate.net/publication/221211998_SVM_Classification_of_Neonatal_Facial_Images_of_Pain.

[62] Behnood Gholami, Wassim M. Haddad, and Allen R. Tannenbaum. "Relevance vector machine learning for neonate pain intensity assessment using digital imaging". In: *IEEE Transactions on Biomedical Engineering* 57 (6 June 2010), pp. 1457–1466. ISSN: 00189294. DOI: 10.1109/TBME.2009.2039214.

[63] Marilena-Catalina Dragomir, Corneliu Florea, and Valentin Pupezescu. "Automatic Subject Independent Pain Intensity Estimation using a Deep Learning Approach". In: *2020 International Conference on e-Health and Bioengineering (EHB)*. IEEE, Oct. 2020, pp. 1–4. ISBN: 978-1-7281-8803-4. DOI: 10.1109/EHB50910.2020.9280190. URL: https://ieeexplore.ieee.org/document/9280190/.

[64] Xuwu Xin et al. "Pain expression assessment based on a locality and identity aware network". In: *IET Image Processing* 15 (12 Oct. 2021), pp. 2948–2958. ISSN: 1751-9667. DOI: 10.1049/IPR2.12282. URL: https://onlinelibrary.wiley.com/doi/full/10.1049/ipr2.12282https://onlinelibrary.wiley.com/doi/abs/

10.1049/ipr2.12282https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/ipr2.12282.

[65] Pau Rodriguez et al. "Deep Pain: Exploiting Long Short-Term Memory Networks for Facial Expression Classification". In: *IEEE Transactions on Cybernetics* 52 (5 May 2022), pp. 3314–3324. ISSN: 21682275. DOI: 10.1109/TCYB.2017.2662199.

[66] Vedhas Pandit et al. "I see it in your eyes: Training the shallowest-possible CNN to recognise emotions and pain from muted web-assisted in-the-wild video-chats in real-time". In: *Information Processing Management* 57 (6 Nov. 2020), p. 102347. ISSN: 0306-4573. DOI: 10.1016/J.IPM.2020.102347.

[67] Patrick Lucey et al. "Automatically Detecting Pain Using Facial Actions". In: *International Conference on Affective Computing and Intelligent Interaction and workshops : [proceedings]. ACII (Conference)* 2009 (Dec. 2009), p. 1. DOI: 10.1109/ACII.2009.5349321. URL: /pmc/articles/PMC3296481//pmc/articles/PMC3296481/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC3296481/.

[68] Ahmed Bilal Ashraf et al. "The painful face – Pain expression recognition using active appearance models". In: *Image and Vision Computing* 27 (12 Nov. 2009), pp. 1788–1796. ISSN: 0262-8856. DOI: 10.1016/J.IMAVIS.2009.05.007.

[69] Ghazal Bargshady et al. "The modeling of human facial pain intensity based on Temporal Convolutional Networks trained with video frames in HSV color space". In: *Applied Soft Computing* 97 (Dec. 2020), p. 106805. ISSN: 15684946. DOI: 10.1016/j.asoc.2020.106805. URL: https://linkinghub.elsevier.com/retrieve/pii/S1568494620307432.

[70] Goyani M Mahesh and Mahesh M Goyani. *Judgmental Feature Based Facial Expression Recognition Systems and FER Datasets-A Comprehensive Study*. Tech. rep. 2019. URL: https://www.researchgate.net/publication/338208229.

[71] Ekman P. and Friesen W. V. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. 1978.

[72] Mohammad Tavakolian, Miguel Bordallo Lopez, and Li Liu. "Self-supervised pain intensity estimation from facial videos via statistical spatiotemporal distillation". In: *Pattern Recognition Letters* 140 (Dec. 2020), pp. 26–33. ISSN: 01678655. DOI: 10.1016/j.patrec.2020.09.012. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167865520303457.

[73] Ehsan Othman et al. "Cross-database evaluation of pain recognition from facial video". In: *International Symposium on Image and Signal Processing and Analysis, ISPA* 2019-September (Sept. 2019), pp. 181–186. ISSN: 18492266. DOI: 10.1109/ISPA.2019.8868562.

[74] Laduona Dai, Joost Broekens, and Khiet P. Truong. "Real-time pain detection in facial expressions for health robotics". In: *2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos, ACIIW 2019* (Sept. 2019), pp. 277–283. DOI: 10.1109/ACIIW.2019.8925192.

[75] Tadas Baltrusaitis et al. "OpenFace 2.0: Facial behavior analysis toolkit". In: *Proceedings - 13th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2018* (June 2018), pp. 59–66. DOI: 10.1109/FG.2018.00019.

[76] Kenneth M. Prkachin and Patricia E. Solomon. "The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain". In: *Pain* 139 (2 Oct. 2008), pp. 267–274. ISSN: 0304-3959. DOI: 10.1016/j.pain.2008.04.010. URL: https://journals.lww.com/00006396-200810150-00006.

[77] J Lawrence et al. "The development of a tool to assess neonatal pain." In: *Neonatal network : NN* 12 (6 Sept. 1993), pp. 59–66. ISSN: 0730-0832.

[78] S. W. Krechel and J. Bildner. "CRIES: a new neonatal postoperative pain measurement score. Initial testing of validity and reliability". In: *Paediatric anaesthesia* 5 (1 1995), pp. 53–61. ISSN: 1155-5645. DOI: 10.1111/J.1460-9592.1995.TB00242.X. URL: https://pubmed.ncbi.nlm.nih.gov/8521311/.

[79] Ruth V.E. Grunau and Kenneth D. Craig. "Pain expression in neonates: facial action and cry". In: *Pain* 28 (3 1987), pp. 395–410. ISSN: 0304-3959. DOI: 10.1016/0304-3959(87)90073-X. URL: https://pubmed.ncbi.nlm.nih.gov/3574966/.

[80] Shannon Fuchs-Lacelle and Thomas Hadjistavropoulos. "Development and preliminary validation of the Pain Assessment Checklist for Seniors with Limited Ability to Communicate (PACSLAC)". In: *Pain Management Nursing* 5 (1 2004), pp. 37–49. ISSN: 15249042. DOI: 10.1016/j.pmn.2003.10.001. URL: https://pubmed.ncbi.nlm.nih.gov/14999652/.

[81] Patricia Lane et al. "A pain assessment tool for people with advanced Alzheimer's and other progressive dementias". In: *Home healthcare nurse* 21 (1 2003), pp. 32–37. ISSN: 0884-741X. DOI: 10.1097/00004045-200301000-00007. URL: https://pubmed.ncbi.nlm.nih.gov/12544460/https://pubmed.ncbi.nlm.nih.gov/12544460/?dopt=Abstract.

[82] Michael R. Villanueva et al. "Pain Assessment for the Dementing Elderly (PADE): reliability and validity of a new measure". In: *Journal of the American Medical Directors Association* 4 (1 2003), pp. 1–8. ISSN: 1525-8610. DOI: 10.1097/01.JAM.0000043419.51772.A3. URL: https://pubmed.ncbi.nlm.nih.gov/12807590/https://pubmed.ncbi.nlm.nih.gov/12807590/?dopt=Abstract.

[83] Mohammad Tavakolian, Carlos Guillermo Bermudez Cruces, and Abdenour Hadid. "Learning to detect genuine versus posed pain from facial expressions using residual generative adversarial networks". In: *Proceedings - 14th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2019* (May 2019). DOI: 10.1109/FG.2019.8756540.

[84] Safaa El Morabit et al. "Automatic Pain Estimation from Facial Expressions: A Comparative Analysis Using Off-the-Shelf CNN Architectures". In: *Electronics* 10 (16 Aug. 2021), p. 1926. ISSN: 2079-9292. DOI: 10.3390/electronics10161926. URL: https://www.mdpi.com/2079-9292/10/16/1926.

[85] Sherly Alphonse, S. Abinaya, and Nishant Kumar. "Pain assessment from facial expression images utilizing Statistical Frei-Chen Mask (SFCM)-based features and DenseNet". In: *Journal of Cloud Computing* 13 (1 Sept. 2024), p. 142. ISSN: 2192-113X. DOI: 10.1186/s13677-024-00706-9. URL: https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-024-00706-9.

[86] Ashish Semwal and Narendra D. Londhe. "Automated facial expression based pain assessment using deep convolutional neural network". In: *Proceedings of the 3rd International Conference on Intelligent Sustainable Systems, ICISS 2020* (Dec. 2020), pp. 366–370. DOI: 10.1109/ICISS49785.2020.9316099.

[87] Philipp Werner, Ayoub Al-Hamadi, and Robert Niese. "Pain recognition and intensity rating based on Comparative Learning". In: *Proceedings - International Conference on Image Processing, ICIP* (2012), pp. 2313–2316. ISSN: 15224880. DOI: 10.1109/ICIP.2012.6467359.

[88] Hugo Touvron et al. "Training data-efficient image transformers & distillation through attention". In: *International Conference on Machine Learning*. Jan. 2020.

[89] Zhun Zhong et al. "Random Erasing Data Augmentation". In: *ArXiv* (Nov. 2017). URL: http://arxiv.org/abs/1708.04896.