UNIVERSITY OF CALIFORNIA,
IRVINE

An End-to-End Platform for Multi-Modal Machine Learning Affective Computing Services

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computer Science

by

Emad Kasaeyan Naeini

Dissertation Committee:
Distinguished Professor Nikil Dutt, Chair
Associate Professor Amir M. Rahmani
Professor Fadi Kurdahi

2022

DEDICATION

I dedicate this thesis to my lovely parents and brother for always being there to support me, motivate me, and give me sincere advice. Undeniably, I owe all my academic achievements to them, and this dedication is the smallest gratitude that I could express to them.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor Prof. Nikil Dutt for his consistent guidance from the beginning of this project up to its last moments. His unwavering assistance, indispensable advice, and technical instructions showed me the proper direction to excel in this project. My sincere gratitude goes to him because he devotedly cooperated in our publication.

I would also like to thank my co-advisor Prof. Amir M. Rahmani not only for his priceless time and review, but also the brilliant opinions through our meetings which have been undoubtedly effective.

My appreciation also extends to committee member, Prof. Fadi Kurdahi for his participation in the thesis committee and providing helpful feedback, which directly or indirectly contributed to this project.

I would like to shout out to my colleagues at DRG and HST labs, Majid Shoushtari, Bryan Donyanavard, Tiago M*ü*ck, Chen-Ying Hsieh, Kasra Moazzemi, Hamid Nejatollahi, Delaram Amiri, Sina Shahhosseini, Sina Labbaf, Biswadip Maity, Ajan Subramanian, Seyed Amirhossein Aqajari, Milad Asgari Mehrabadi, Rui Cao, Kenneth Stewart, Saehanseul Yi, Caio Batista de Melo, Khuong Vo, Amirhosein Mohaddesi, and Dongjoo Seo for their friendship and for sharing the pain.

Finally, my work would not have been possible without funding from the Electrical Engineering and Computer Science department or permission from the Institute of Electrical and Electronics Engineers (IEEE), Association for Computing Machinery (ACM), and Journal of Medical Internet Research (JMIR) to include content from my previously published work in [78, 79, 8, 30, 90, 10, 112].

VITA

Emad Kasaeyan Naeini

EDUCATION

Doctor of Philosophy in Computer Engineering **2022**
University of California, Irvine *Irvine, CA*

Master of Science in Electrical and Computer Engineering **2020**
University of California, Irvine *Irvine, CA*

Bachelor of Science in Electrical Engineering **2017**
Sharif University of Technology *Tehran, Iran*

RESEARCH EXPERIENCE

Graduate Student Researcher **2017–2022**
University of California, Irvine *Irvine, California*

TEACHING EXPERIENCE

Teaching Assistant **2018–2022**
University name *Irvine, California*

REFEREED JOURNAL PUBLICATIONS

Prospective Study Evaluating a Pain Assessment Tool in a Postoperative Environment: Protocol for Algorithm Testing and Enhancement **Jul 2020**
JMIR Research Protocols (JRP)

Pain Recognition With Electrocardiographic Features in Postoperative Patients: Method Validation Study **May 2021**
Journal of Medical Internet Research (JMIR)

Pain Assessment Tool With Electrodermal Activity for Postoperative Patients: Method Validation Study **May 2021**
JMIR mHealth and uHealth (JMU)

REFEREED CONFERENCE PUBLICATIONS

pyEDA: An open-source python toolkit for pre-processing and feature extraction of electrodermal activity **Jan 2021**
Ambient Systems, Networks and Technologies (ANT)

Objective Pain Assessment Using Wrist-based PPG Signals: A Respiratory Rate Based Method **Nov 2021**
International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)

AMSER: Adaptive Multimodal Sensing for Energy Efficient and Resilient eHealth Systems **Mar 2022**
Design, Automation & Test in Europe Conference & Exhibition (DATE)

SOFTWARE

pyEDA `https://github.com/HealthSciTech/pyEDA`
Python Library for Electrodermal Activity Analysis

pyPPGqa `https://github.com/HealthSciTech/pyPPGqa`
Python Library for PPG Signal Quality Assessment

ABSTRACT OF THE DISSERTATION

An End-to-End Platform for Multi-Modal Machine Learning Affective Computing Services

By

Emad Kasaeyan Naeini

Doctor of Philosophy in Computer Science

University of California, Irvine, 2022

Distinguished Professor Nikil Dutt, Chair

Smart affective computing applications deliberately influence pain, emotion and other affective phenomena, and are fundamental to human experience, health and well-being. Affective states such as pain, stress, and emotion are intrinsically subjective in nature, posing challenges for objective assessment and quantification of such affective phenomena. Prior efforts in affective computing provide a foundation for the automated analysis of affective states, but still face challenges in real-life, everyday settings. We use pain as an exemplar for affective computing services. Pain assessment is critical for optimal treatment, and is particularly important during periods of acute pain since inadequately treated acute pain increases the risk of chronic pain. Historically, patients have served as the main assessment tool as they are able to self-report their pain presence and severity on standard, but somewhat subjective pain scales. However, it remains a challenge to assess pain from patients who cannot self-report. Automatic pain recognition systems could be crucial to facilitating accurate, objective, and real-time pain measurement, that can in turn improve pain management and ultimately lead to improved patient outcomes. Using pain as an exemplar of smart affective computing services, this thesis proposes the use of multimodal sensing of physiological and behavioral input data, transmits the data to edge and/or cloud nodes, and processes data with compute-intensive machine learning (ML) algorithms. The efficiency of ML-driven applications for affective computing (e.g., pain assessment) is greatly affected by run-time variations resulting from continuous stream of noisy input data, unreliable network connections, and the variations in computational requirements of ML algorithms. Towards that end, this thesis evaluates and automates objective and real-time multimodal pain assessment algorithms, and performs design space exploration of accuracy-performance-energy trade-offs and sense-compute co-optimization for multimodal machine learning (MMML) methods. The approach developed in this thesis could be used for the design and development of an end-to-end platform for multimodal machine learning affective computing services, together with a thorough analysis of their roles in the prediction, performance and energy in future studies.

Chapter 1

Introduction

Smart affective computing applications that deliberately influences emotion or other affective phenomena are fundamental to human experience, health and well-being. The connection between emotional states and physical health has become more known and has motivated the field of affective computing. Affective computing uses both hardware and software technology to detect the affective state of a person. Affective states such as pain, stress, and emotion are subjective in nature which are beyond sensory feeling combining affective factors. As a specific instance, physical pain is closely related to emotional states that may modulate the experience of pain and vice versa. Furthermore, principles and techniques from affective computing provide a solid basis for the automated analysis of pain-related states. Smart affective computing services integrate remote sensing, continuous monitoring, wireless transmission, data analytics, and machine learning to deliver intelligent patient-centric digital healthcare and well-being services [84]. These applications are particularly effective for managing chronic patients through continuous monitoring, extracting clinically relevant data with minimal intrusion [43]. In this work, I seek to develop automatic methods for objectively quantifying and managing pain as an exemplar case study of affective computing services.

1.1 Overview

Pain is a single major reason for people seeking medical care and is associated with many illnesses [183]. Uncontrolled pain places patients at risk for numerous adverse psychological and physiological consequences, some of which may be life-threatening [172]. Pain assessment is critical to the optimal treatment of pain. At present, there is a wide variability in how pain is assessed and managed at bedside, and the prevalent practices remain suboptimal [57]. Inadequately treated pain has major physiological, psychological, economic, and social ramifications for patients, their families, and the society [12]. Under-treatment of pain could result in many adverse effects and other complications, and may evolve into chronic pain syndromes. It could also cause delayed discharge or prolonged recovery, which may incur higher healthcare costs and more patient suffering [172]. Overtreatment of pain, on the other hand, may result in unintended adverse consequences such as acute respiratory complications or long term complications such as opioid addiction. These issues are particularly pronounced for non-communicative patients who are unable to articulate their experience of pain [18]. Pain assessment mainly refers to an assessment of pain intensity which helps to decide the type of intervention that will be used including the type of analgesic to be administered and the dosage [193]. In addition to intensity, the location and quality (e.g., aching and burning) of pain are also the aspects of assessment.

Pain is "an unpleasant sensory and emotional experience associated with actual or potential tissue damage, or described in terms of such damage" [69], according to the International Association for the Study of Pain (IASP). Pain may be classified as either acute or chronic; Kent *et al.* [81] described acute pain as encompassing the immediate, time-limited bodily response to a noxious stimulus that triggers actions to avoid or mitigate ongoing injury. Chronic pain was first defined loosely by Bonica [24] as pain that extends beyond an expected time frame; currently, chronic pain is defined as "persistent or recurrent pain lasting longer than three months" [185]. The "gold standard" of pain assessment is self-reporting.

Pain intensity is assessed with a pain scale in one of several forms. For example, in acute postoperative pain, a score 4 in the 11-point scale from 0 to 10 is clinically important borderline of receiving adequate pain control [93, 110]. Pain assessment documentation can be varied among various patients population in consonance with their behavior, e.g., for postoperative acute pain patients the assessment could be regularly every 4 hours plus 1 hour after the intervention [182], or every 15 minute after the surgery [67]. Thus, pain as a multivalent, dynamic, and ambiguous phenomenon is difficult to quantify [139] (e.g., during critical illness, infants and preverbal toddlers, patients under sedation or anesthesia, persons with intellectual disabilities, patients at the end of life) [27]. The alternative for these cases is through caregivers using pain behavior observation tools. Common behavioral responses to pain are facial expression, body movements, and vocalization. Additionally, some physiological signals such as heart rate, heart rate variability, systolic blood pressure, and respiration rate are considered nonverbal signals of pain. Automated and continuous pain assessment for poorly communicating patients can enable timely treatment, reduce the monitoring burden on clinicians, and contribute to optimizing the use of analgesics and managing side effects and complications [122].

Researchers have attempted to develop objective pain assessment tools through analysis of physiological pain indicators, such as heart rate (HR), heart rate variability (HRV), and electrodermal activity (EDA) [11, 168, 63, 53]. However, pain assessment by using only these signals can be unreliable, as there are various other factors that alter these vital signs [39]. Objective pain assessment using behavioral signs such as facial expression has recently gained attention [89, 140]. Several techniques in this category, such as those using computer vision [165, 100, 76, 75], suffer from feasibility issues in clinical settings due to privacy and practical barriers to deployment. Recently, facial electromyography (EMG) has proven to be viable in detecting facial expressions due to the fact that the amplitude of frontal muscle activity during sedation and anesthesia increases due to painful stimuli [58, 40]. However, these solutions to date have only a) focused on the data analytics aspect of pain assessment, and b) been evaluated in healthy volunteers. The automatic assessment tool is expected to work as accurately and reliably as self-report, and at least as well as human experts. It is thus imperative to develop an objective pain assessment tool to improve the well-being and care processes of noncommunicative patients. Such a tool can also benefit other patient populations with more accurate assessment and more timely treatment. This thesis aims to reach this ultimate goal. The study was initialized and conducted within the iHurt Pain Assessment research group.

The biopotentials included in this study are several physiological signals including:

- Electromyogram (EMG) - the electrical activity produced by facial muscles which are also a pain behavior indicator corresponding to facial expression.
- Electrocardiogram (ECG) - the electrical activity of the cardiac cycle
- Electrodermal Activity (EDA) - the skin resistance change due to the activity of sweat glands in the skin,
- Photoplethysmogram (PPG) - the optical measurement of the variation of blood flow

Additionally, the PPG-based respiratory rate is also part of the analysis. The responses to postoperative acute pain through these biopotentials were recorded and analyzed.

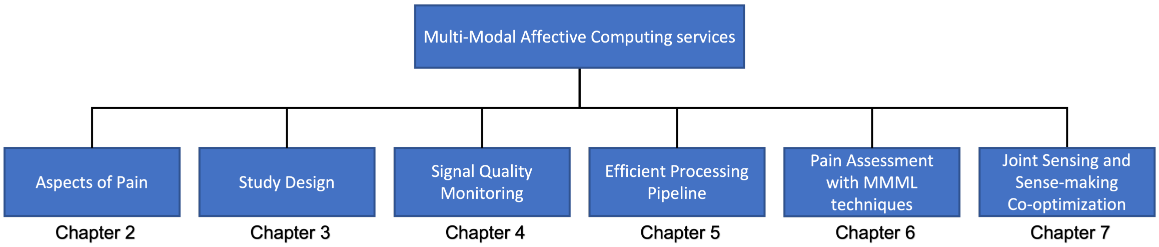

Multi-Modal Affective Computing services
Aspects of Pain
Chapter 2
Study Design
Chapter 3
Signal Quality Monitoring
Chapter 4
Efficient Processing Pipeline
Chapter 5
Pain Assessment with MMML techniques
Chapter 6
Joint Sensing and Sense-making Co-optimization
Chapter 7

Figure 1.1: Structure of the thesis

1.2 Research aim and objectives

The research aim of this dissertation is two-fold.

One is developing and evaluating an automatic and versatile pain assessment tool in a reliable and objective way for noncommunicative patients to improve the current standard of self-reported pain assessment in clinical settings, which can process and relate information from physiological and behavioral signs. We posit that in objective pain assessment, the captured interactions between modalities are both *supplementary* and *complementary*. In other words, the information in different modalities can improve the robustness of the multimodal models as well as enhance the prediction performance in noisy scenarios. The resultant solution will be an automatic, user-centered, and versatile pain assessment system, based on the Internet of Things (IoT) [108]. Wearable technology is a promising paradigm to integrate several technologies and communication solutions [163, 77].

The other one is improving the energy efficiency and resiliency of multi-modal machine learning (MMML) affective computing services by monitoring input signal quality and discrepancy detection and bringing adaptive and intelligent control to a multi-modal sensor-edge platform to optimize response time, and energy provided accuracy requirements.

Previous research studies on experimental pain on healthy volunteers led us to a better understanding of the system design and answered questions regarding reliability, validity, and the limitations of the approach. Building on this knowledge we are able to further develop and research a pain assessment tool in post-operative patients likely experiencing mild to moderate pain.

1.3 Thesis contributions

The main contributions of this dissertation are outlined below.

iHurt Pain Database

We acquired a new set of data collected from postoperative patients having real pain likely experiencing mild to moderate pain as opposed to experimental stimulated pain. To develop and validate pain assessment tools, it is crucial to record data of people experiencing pain. This database consists of multiple sensory input data of 25 patients from face- (EMG), chest- (ECG), and wrist-worn electrodes and devices (EDA, PPG, Accelerometer).

Pain Assessment Methods

We developed several ML-based automatic and objective pain assessment method using unimodal and multimodal models based on physiological and behavioral pain indicators. We conducted several experiments among four different pain intensities vs baseline levels of pain. Models for each of these intensities were varied based on the modalities, different types of data labeling augmentation techniques, machine learning algorithms, and the type of modality fusion used.

Intelligent and Adaptive Framework in MMML Affective Computing Services

We proposed two joint sensing and sense-making approaches that embeds signal quality monitoring, adaptive sensing and an intelligent sense-making mechanism for qualitative assessment of data and control flows in MMML services. We designed a sensor-edge framework for multi-modal machine learning based affective computing applications, capable of monitoring input signal quality, and detection of discrepancies to guide sensing and sense-making optimization decisions. We designed a reinforcement learning orchestration scheme for affective computing services on sensor-edge networks. The orchestrator performs adaptive feature and modality selection, machine learning model selection, and sensor configuration based on sensor modality monitoring at runtime to optimize energy consumption given accuracy requirements.

1.4 Organization

The rest of thesis as shown in Figure 1.1 is organized as follows:

- **Chapter 2:** provides an introduction to an affective state – pain – as an exemplar case study of affective computing domain, its assessment scales, mechanisms and responses, and reviews prior research works for automated pain assessment systems.
- **Chapter 3:** presents the study design for validation of the system in a clinical setting and iHurt Pain Database specifications. The details of the study design are the study protocol, patients demographic characteristics, and biosignal measurements. The collected data and signals are summarized and listed.
- **Chapter 4:** proposes a deep learning-based quality assessment technique for PPG signal and introduces an automatic annotation method to develop PPG signal quality assessment.
- **Chapter 5:** presents a complete data processing pipeline for each of the biosignals collected within the data acquisition phase. Various number of handcrafted features in different domains and automatic features are derived as a result of signal denoising and feature extraction.

- **Chapter 6:** proposes augmentation methods and introduces unimodal and multimodal machine learning models for the assessment of pain from physiological signals.

- **Chapter 7:** proposes an intelligent and adaptive multimodal sensing framework for energy efficient and resilient affective computing services.

- **Chapter 8:** concludes this work, discuss existing strength, limitations and alternative strategies, and provide future research directions.

Chapter 2

Related Works and Background

This chapter gives a comprehensive background knowledge on various aspects of pain from Figure 1.1 as an exemplar of affective computing domain including current approaches to assess pain, pain mechanisms and responses. The mechanisms and responses indicate labels of pain and potential pain indicators and the materials for learning, respectively, when using machine learning methods.

2.1 Background

2.1.1 Subjective Nature of Pain

Pain is “an unpleasant sensory and emotional experience associated with actual or potential tissue damage, or described in terms of such damage” [69], according to the most widely accepted definition. Pain is not only a sensory phenomenon and the complex sequences of behavior that characterizes pain are determined by sensory, motivational, and cognitive components. Pain is considered to be a subjective experience that is related to each individual in early life through experiences related to injury [14]. Such pain, which is termed acute pain, usually lasts hours, days, or weeks. Acute pain is associated with soft tissue damage, a surgical procedure, or a brief disease process and fosters avoidance of the harmful action in the future and promotes healing by inhibiting activities that might cause further tissue damage [196]. Pain, as a susceptible and ambiguous phenomenon, is difficult to quantify [139], particularly when the patient's own opinion is difficult to reach due to their limited ability to communicate, as in patients under sedation or anesthesia, persons with intellectual disabilities, infants, and patients during critical illness [28].

In clinical trails and pain management patient self-report is the gold standard pain assessment tool. Moreover, patient behaviors are observed as pain indicators and various behavioral pain assessment tools have been designed for each population of patients in clinical practice. Unfortunately, due to difference of patient population and given context no behavioral pain assessment tool applies to all. The reliability and validity of a tool should be ensured in each case.

Considering the complex nature of pain, the scope of this study is to realize the relations between physiological signals and intensity of acute pain derived from postoperative patients to predict different pain intensity levels. This will promote advancements in both observational and physiological pain measurement. Uncontrolled acute pain could cause some serious complications and may evolve into chronic pain. This could cause longer recovery in hospitals and delayed discharge, higher health care costs, and major psychological, financial, and social ramifications for patients [173]. However, overtreatment of pain can also result in adverse effects such as hospital readmission due to poorly controlled pain after discharge or long-term opioid dependence.

2.1.2 Pain Assessment Scales/Tools

Acute pain is a common experience in the post-anesthesia care unit (PACU) in the immediate period following surgery. According to Chou *et al.* [36], pain occurs in 80% of patients following surgery and 75% of patients with pain report their pain as either moderate, severe, or extreme. In clinical practice, pain intensity is commonly used to characterize pain through patient's self-report according to severity, sensory quality, location, temporal features, and factors that alleviate and intensify pain. Self-report refers to conscious communication of pain-related information by the person in pain.

In the research on automatic pain intensity recognition, the existing pain assessment tools and concepts act as the ground truth in the system development and validation. As pain intensity is difficult to quantify [139], the criterion standard of pain assessment in the PACU is self-report, as pain is a subjective experience. There exists different categories of pain intensity scales, tools such as a numeric rating scale (NRS) and visual analog scale (VAS) [181], the two most well-known pain intensity scales. NRS approach of self-report allow patients to have the option to verbally rate their pain intensity from 0 to 10 where zero indicates the absence of pain, while 10 represents most intense pain possible. VAS, however, is a continuous form of indicating pain intensity from 0 (no pain) to 10 (worst imaginable pain) usually on a 100mm in length, anchored by word descriptors at each end, ”no pain” and ”worst pain imaginable”. Patients can mark a position on this line to report their level of pain. Third scale of self-report, Verbal Rating Scale (VRS), use discriminative verbal categories, such as: no pain, mild pain, moderate pain, severe pain, and unbearable pain. Depending on the situation, various pain scales may be preferred. For instance, both VRS and NRS can be easily setup without paper and pencil, whereas VAS offers more quantitative pain intensity differentiation. In addition, there exist other scales for patient populations unable to self-report. For this population, the pain scores of some behavioral tools are derived from adding up the scores of several indicators. Therefore, the range of the scale could be different from the standard pain intensity rating. Numerous scales have been designed and validated for various populations, such as infants and preverbal toddlers (e.g., NIPS, CRIES, FLACC) [199]; elderly people with severe dementia, (e.g., PACSLAC, DOLOPLUS2, PAINAD) [204]; and critically ill and/or unconscious subjects (e.g., BPS, CPOT, NVPS) [68]. Most of these scales consider facial expression, body movements, muscle tension, while some include vital parameters.

2.1.3 Pain Mechanisms and Responses

Pain is a subjective experience generated by the brain to protect ourselves. Pain observation tools, pain intensity recognition approaches via measurable signals, and pain reported by parents or family members are proxy measures and are objective in essence. Pain is not only a sensory phenomenon and the complex sequences of behavior that characterizes pain are determined by sensory, motivational, and cognitive components. Pain is characterized by severity, location, duration, and quality; it is unpleasant and motivates activity for relief; and it is influenced by cognitions such as evaluation of an seriousness injury. Pain experience must be carefully distinguished from the pain cause (such as tissue injury), the pain response (verbal and non-verbal manifestations), and pain assessment (e.g., by a caregiver). The pain cause is often diagnosable (e.g., a fracture) and it may be controlled in deliberate pain stimulation (e.g., neurological assessments), but it may also be unknown or absent (especially in chronic pain).

Typically, pain originates from noxious stimuli, e.g., due to tissue injury, that lead to a response of the sensory nervous system called nociception. Pain experience is modulated by personal and inter-personal factors, e.g., cognition, past experience, and situation [115], [125], [126]. As a result, the same stimulus may lead to different pain experiences. In rare cases, people do not experience any pain, and this brings them harm [127]; however, pain usually causes observable pain responses, which are modulated by personal and contextual factors. Pain responses may be categorized in physiological responses, and behavioral responses.

Physiological Responses

Pain sensation through an extensive interactions between neural structures induces alterations in the sympathetic nervous system, resulting in measurable changes in various physiological signals [171]. The signals reflecting Autonomic Nervous System (ANS) activities are listed in Table 2.1 together with some Nociception/Antinociception balance indexes. To this date, research in the estimation of pain intensity has mainly focused on physiological features. These signals are reviewed in the following part of this subsection.

Electrodermal Activity

Electrodermal activity (EDA) monitors electrical conductance in the skin due to the autonomic activation of sweat glands in the skin. As EDA reflects activity only within the sympathetic activity, the increased sympathetic outflow associated with pain causes sweat to be discharged into pores on the skin surface until the sweat is reabsorbed or evaporated [22]. EDA is also referred to as skin conductance, or galvanic skin response (GSR).

Inter-Beat Interval

Sympathetic and parasympathetic activities are traceable from the cardiovascular regulation. Increase in sympathetic activity affects the *heart rate*, leading to tachycardia, and *heart rate variability* (HRV), an index of autonomic regulation of heart rate. The inter-beat intervals are extracted from electrocardiography (ECG) mostly and from PPG in some cases. The normal inter-beat intervals are called normal-to-normal (NN) series.

Further HRV analysis can be made to trace sympathetic or parasympathetic activity within a time period in various domains such as time, frequency and in other forms. HRV analysis can be on an ultra-short term (less than a minute), short-term (1-5 minute), and long term (at least 24hours) periods depending on the study design.

Blood Pressure

Pain increases peripheral vascular resistance and stroke volume which among with increased heart rate leads to an elevation in resting blood pressure. Also, in the CARDEAN index [147], the change in continuous systolic blood pressure is entangled with he change of NN series.

Photoplethysmography Amplitude

The up or down of PPG amplitude (PPGA) can be regulated by anesthesia, sympathetic activation, arterial blood pressure (ABP) increase and some other factors. The sympathetic activation could lower PPG, while anesthetics could increase PPGA, and the rise of ABP may lead to either PPGA up or down due to different causes [88].

Pupil Diameter

Under dual sympathetic and parasympathetic control (dilating and constricting the pupil, respectively), due to the pupil dilation reflex, pain has impacts on pupil diameter [32].

Electroencephalography

Pain affects the electrical activity of brain cortical regions [184]. Electroencephalography (EEG) has shown promise to detect changes in electrical activity in the brain cortex thereby detecting patterns of response to pain [98].

Respiration rate

Breathing slowly and paced slow deep breathing are some routines in the hospital for pain relief, although the physiological mechanisms behind it are not fully known yet. Recently, a number of experimental and clinical studies have suggested that pain influences respiration by increasing its frequency, flow, and volume [80, 51].

Table 2.1: The physiological responses to pain

Signal	Index name	Parameter(s)	ANS activity	Model
EDA	skin conductance	number & amplitude of SCR fluctuations (NSCF & ASCF)	sympathetic	painful stimuli induce an immediate increase in peaks per second
PPG	Surgical pleth index (SPI)	pulse wave amplitude (PPGA) and heart beat interval (HBI)	sympathetic	SPI=100 - (0.7×PPGA$_{norm}$+0.3 × HBI$_{norm}$)
Pupil dilation	pupillometry dilatation reflex	pupillary diameter and the light-induced pupillary	sympathetic	-
ECG	Analgesia/Nociception index (ANI)	0.15Hz-0.4Hz bandpassed RR series$_{norm}$	parasympathetic	ANI=100×(5.1×AUC$_{min}$ +1.2)/12.8 , where AUC is the area between the envelope of local maxima and local minima
Blood Pressure	CARdiovascular DEpth of ANalgesia (CARDEN)	RR under curve and systolic blood pressure under curve	sympathetic	a minor elevation in blood pressure followed by minor tachycardia

Behavioral Responses

All behavioral responses to pain are in pursuit of two main functions: 1) To protect our own bodies, "pain grabs attention, interrupts associated behavior, and urges action towards mitigating it" [196], such as reflexive withdrawal of the hand from a hot surface. 2) To communicate pain through showing need for help to potential caregivers and hiding vulnerabilities from antagonists–a behavior that probably developed since it increased chances of survival and reproduction [196]. Common behavioral responses to pain are facial expression, body movements, and vocalization. As non communicative subjects are the patient population for this study, behavioral signs of pain assessment according to [68] can be observed through tools such as 1) BPS: Behavioral Pain Scale [123], 2) CPOT: Critical-Care Pain Observation Tool [51], 3) FLACC: Face, Legs, Activity, Cry, and Consolability Behavioral Assessment Tool [191], 4) PBAT: Pain Behavioral Assessment Tool [137], 5) NPAT: Nonverbal Pain Assessment Tool [86], 6) NVPS: Nonverbal Pain Scale [73], and 7) BPAT: 8-item Behavior Pain Assessment Tool [52].

In most of these tools, the output is a final score from a sum of sub-scores from each category. In each category/indicator/item, the sub-score is defined by either the presence or degree of the behavior. A larger score of each tool may indicate higher pain intensity, however, the score number and pain intensity number are not highly correlated. For example, it is reported that BPAT showed a moderate ability to discriminate severe levels of pain intensity (NRS$\geq$8), and there is a moderate correlation between pain distress and behavioral scores during common procedures performed in ICU patients which also supports the interrelation between the affective and behavioral dimensions of pain [52].

Facial Expression

Pain is associated with couple of facial expressions specifically that occur relatively consistently across a range of clinical pain conditions and experimental pain modalities. In

Table 2.2: The description of facial expression at different pain scores

Facial action unit		Muscular basis	Description Scale			
No.	Facial action		Grimace	Wince	PSPI	CPOT
AU4	brow lowering	Corrugator supercilii	✓		✓	✓
AU6	cheek raising	Outer Orbicularis oculi	✓	✓	✓	✓
AU7	tightening eyelids	Inner Orbicularis oculi	✓	✓	✓	✓
AU9	wrinkling of nose	Levator labii superioris			✓	✓
AU10	raising of upper lip	Levator labii superioris			✓	✓
AU12	pulling at corner lip	Zygomaticus major				
AU20	stretching lips	Risorius	✓			
AU25	parting lips	Depressor Labii	✓			
AU43	closing eyes	Relaxation of Levator	✓		✓	✓

tools such as BPS, CPOT, and NPAT, the presence of some facial expressions determines their score number, whereas in FLACC and NVPS the concern is more about the frequency of shown facial expressions. Table 2.3 summarizes the pain scores in various behavioural pain assessment tools based on facial expression. Rising intensities of noxious stimulation will cause increase in magnitude of facial movements. Facial expression research is usually conducted using the Facial Action Coding System (FACS), which describes all visually distinguishable facial muscle activity from 44 Action Units (AUs). The AUs involved in pain facial expressions among adults are listed in Table 2.2 and summarized in the review [135]. Based on a research study conducted by Prkachain [134] four actions - brow lowering (AU4), orbital tightening (AU6 and AU7), levator contraction (AU9 and AU10) and eye closure (AU43) - carried the bulk of information about pain. In a recent follow up to this work, Prkachin and Solomon [136] defined a pain scale based on these core action units which can be calculated as in Equation 2.1:

$$Pain = AU4 + Max(AU6|AU7) + Max(AU9|AU10) + AU43 \tag{2.1}$$

AUs are scored on a 6-point intensity scale that ranges from 0 (absent) to 5 (maximum intensity) and binary (0 = absent, 1 = present) for AU43 (Eye closing). Facial expressions can be recorded via sEMG as well, where surface electrodes are placed on the muscle area of interest to capture the electric potential generated by muscle cells during muscle contraction.

Body movements

The body movements in reaction to pain are more versatile in behavioral pain assessment tools compared to facial expressions. These include voluntary movements (e.g., legs movements in FLACC and body movements or activity in CPOT, NVPS and NPAT); protective reflexes (e.g., muscle tension in CPOT and guarding in NPAT); and the posture or the static state of the body (e.g., rigid, clenched fists, and fetal position). There have been quite few research studies on automatic pain assessment with respect to body movements. However, there have been some developments for human-computer interaction through hand and body gesture recognition using a camera. Kessous et al [83] recognize the pattern of emotions with a fusion of facial expression, body gesture, and acoustic analysis. This can be a starting point for future pain assessment studies.

Vocalization

Another pain behavior is vocalizations, such as paralinguistic vocalizations (sighing, moaning, whimpering, sobbing, crying, and screaming), and voice quality aspects such as amplitude and tone are observed during verbal self-report [51]. Vocalizations have been researched mainly as part of the design and validation of observational pain scales such as CPOT [173], and FLACC [68].

Table 2.3: The description of facial expression at different pain scores

Signal	Minimum Score	Middle Score	Maximum Score
BPS	(1) Relaxed	(2) Partially tightened (3) Fully tightened	(4) Grimacing
CPOT	(0) Relaxed, neutral No muscular tension observed	(1) Tense Presence of frowning, brow lowering, orbit tightening, and levator contraction	(3) Grimacing All of the previous facial movements plus eyelid tightly closed
NPAT	(0) Relaxed, calm expression	(1) Drawn around mouth and eyes; tense	(2) Facial frowning, wincing, grimacing
FLACC	(0) No particular expression or smile	Occasional grimace, frown, withdrawn or disinterested	(2) Frequent to constant frown, clenched jaw, quivering chin
NVPS	(0) No particular expression or smile	(1) Occasional grimace, tearing frown or wrinkled forehead	(2) Frequent grimace, tearing, frown or wrinkled forehead

Chapter 3

Study Design and the iHurt Database

Healthy volunteers are commonly involved in the studies to develop an automatic pain assessment tool. To be able to perform a validation on the end-to-end platform first we need to actually have a study design that is done in a clinical setting. This study is a prospective observational data collection from postoperative patients likely having mild to moderate pain. This chapter explains the Study Design block shown in Figure 1.1 in detail with a complete study protocol and will introduce a new database collected from postoperative patients the iHurt Pain database. The signals collected in this database are physiological signals (ECG, EDA, PPG-based respiratory) and one behavioral signal, facial expressions using facial sEMG signals.

3.1 Study Protocol

The prospective study was conducted at UCIMC in Orange, California. The APS unit at the medical center serves approximately 100 patients weekly, enabling the lead Doctor of Medicine to recruit patients. All 25 participants recruited for this study met the following criteria: (1) age at least 18 years, (2) received a consult by the APS, (3) able to communicate, (4) able to provide written informed consent, and (5) healthy, intact facial skin. They were excluded if they had (1) any diagnosed condition affecting cognitive functions (dementia, psychosis), (2) any diagnosed condition affecting the central nervous system, facial nerves or muscles, (3) deformities on hand that prevent sensor placement, or (4) significant facial hair growth in the area where the sensors were going to be attached.

After IRB approval, we screened the medical records at UCIMC, to which APS has access, to determine potential participants eligibility to participate in this study based on the protocol inclusion and exclusion criteria. The anesthesiologist will approach their patients directly about study participation at the University of California Irvine Douglas Hospital, Orange, CA, USA. The study procedure was continued if patients showed interest and were suitable for the study (according to the inclusion and exclusion criteria). The study physician explained the study in detail, providing both oral and written information. If a patient decided not to participate in the study, the study were discontinued for this patient. If the patient was still willing to participate, the study participant enrollment log would be updated accordingly. During the study, participants' experience of pain intensity were recorded using the NRS.

We considered the natural variation in caseload (such as trauma and elective procedures) coupled with the fact that the recovery period is different for each patient. All candidates considered for enrollment were experiencing postoperative pain during their hospitalization, and all were receiving analgesic treatment. The study team ensured that patient safety would not be compromised while maintaining regulatory compliance. Patients got both oral and written information about the details of the study. Candidates were provided at least 24 hours to consider participation in the study before finalizing the consent form.

The data collection took place in a quiet room where powerline interference was avoided as much as possible. One study analyst and one clinic researcher were also in the room to set the signal acquisition system and instruct the study subject, respectively. The clinic researcher briefly introduced the processes of the study to the patient, then the study subject was recruited to participate in this study after obtaining the written consent form. The eight-channel biopotential acquisition device [153] for EMG and ECG monitoring was attached to the study subject. Six lead Facial sEMG electrodes were placed on left hand side facial area. Two lead ECG electrodes were placed on left and right arm. Empatica E4 sensor and a transcutaneous electrical nerve stimulation (TENS) unit were attached to the non-IV arm. After making sure the setup is ready, the recording started.

Approximately 30 minutes of continuous biosignals (EMG, ECG, EDA, and PPG) data was collected from the participants. We separated this 30-minute period into 2 parts: control (baseline pain) and experimental. Each part was consisted of 2 to 3 challenge intervals in an attempt to capture pain perception before, during, and after a stimulus, with appropriate rest periods to make the statistical analysis more powerful.

In the control part, we used a TENS unit to obtain the patient's baseline pain level by placing the TENS on the participant's forearm and consistently prompting for NRS pain scores. We believed it is prudent to provide some level of baseline assessment above the patient's existing postsurgical pain to attempt to find a baseline of pain for comparison among participants. Therefore, we used TENS as a means of standardizing the patient threshold for experiencing pain and as a way to keep the data consistent with the previous phase of the study conducted on healthy volunteers.

In the experimental part, patients were engaged with soft activities (e.g., walking, coughing, sitting, and lifting legs) that could cause a pain sensation. The participant's experience of pain were recorded using NRS. We expected to find solutions from multiple parameters that are robust in response to different acute pain cases or study designs. All protected health information were redacted prior to data analysis.

3.2 Study Design

The study was designed to collect data in two steps: control (baseline pain) and experimental. In the first step of data collection only TENS unit is used to get the baseline of the person. Patient is asked to increase the intensity of the TENS unit till the level that is tolerable for him/her and then hold it there for at least 30s. Finally, patients were asked to decrease it to go back to level 0. This procedure was repeated three times. In the second step, patients did some soft activities such as walking, coughing, seating, lifting legs, etc that could cause them feel pain with the non-invasive devices connected excluding the TENS device. Patients were asked to do a soft activity and hold that position for at least 30s. Then asked to go back to the normal situation. This procedure was repeated three times. Before and after each interval of situation changing the study subject is asked about pain level. NRS and the type of activity was recorded at each time stamp.

3.2.1 Pain Demographic Characteristics

A total of 25 patients with acute pain were engaged by APS and recruited for this study at UCIMC. We removed 3 participants' data from the final dataset due to the presence of excessive motion artifacts. We also excluded 2 additional patients since they were wearing the Empatica E4 watch on their IV arm, which resulted in unreliable EDA signals due to conditions like skin rash and itching. This left us with data from 20 patients to build our pain recognition system. The dataset also contains rich annotation with self-reported pain scores based on the 11-point Numeric Rating Scale (NRS) from 0 – 10.

The average age of patients was 55.6 years (SD 16.24, range 23-89); 52% (13/25) of patients were male and 48% (12/25) of patients were female (Table 3.1). All of the patients (n=20) were taking prescription medication at the time of the study. The nature of the procedures

Table 3.1: Patient demographic characteristics (N=25)

Variable	Value	Range
Patients excluded due to arrhythmia, n (%)	3 (12)	N/A[a]
Patients excluded due to missing ECG data, n (%)	2 (8)	N/A
Gender, male, n (%)	13 (52)	N/A
Weight (kg), mean (SD)	76.56 (17.31)	52.2-112.2
Height (cm), mean (SD)	170.9 (10.44)	152.4-193
BMI[b] (kg/m2), mean (SD)	26.33 (6.14)	15.1-38.73
Procedure domain (n=20), n (%)		
General surgery	10 (50)	N/A
Orthopedics	5 (25)	N/A
Trauma	3 (15)	N/A
Urology	2 (10)	N/A

[a] N/A: Not Applicable

[b] N/A: Body Mass Index

for each participant included the following domains: 50% general surgery (diagnostic laparoscopy, exploratory laparotomy, and vascular), 25% orthopedics, 15% trauma (thoracic pain and rib plating), and 10% urology (cystectomy and bladder augmentation).

3.2.2 Biosignal Acquisition using the iHurt System

iHurt is a system that measures facial muscle activity (i.e., changes in facial expression) in conjunction with physiological signals such as heart rate, heart rate variability, respiratory rate, and electrodermal activity for the purpose of developing an algorithm for pain assessment in hospitalized patients. The system as illustrated in Figure 3.1 uses the two following components to capture raw signals.

Eight-Channel Biopotential Acquisition Device

Our team at the University of Turku, Finland developed a biopotential acquisition device to measure ECG and EMG signals. The device incorporates commercially available elec-

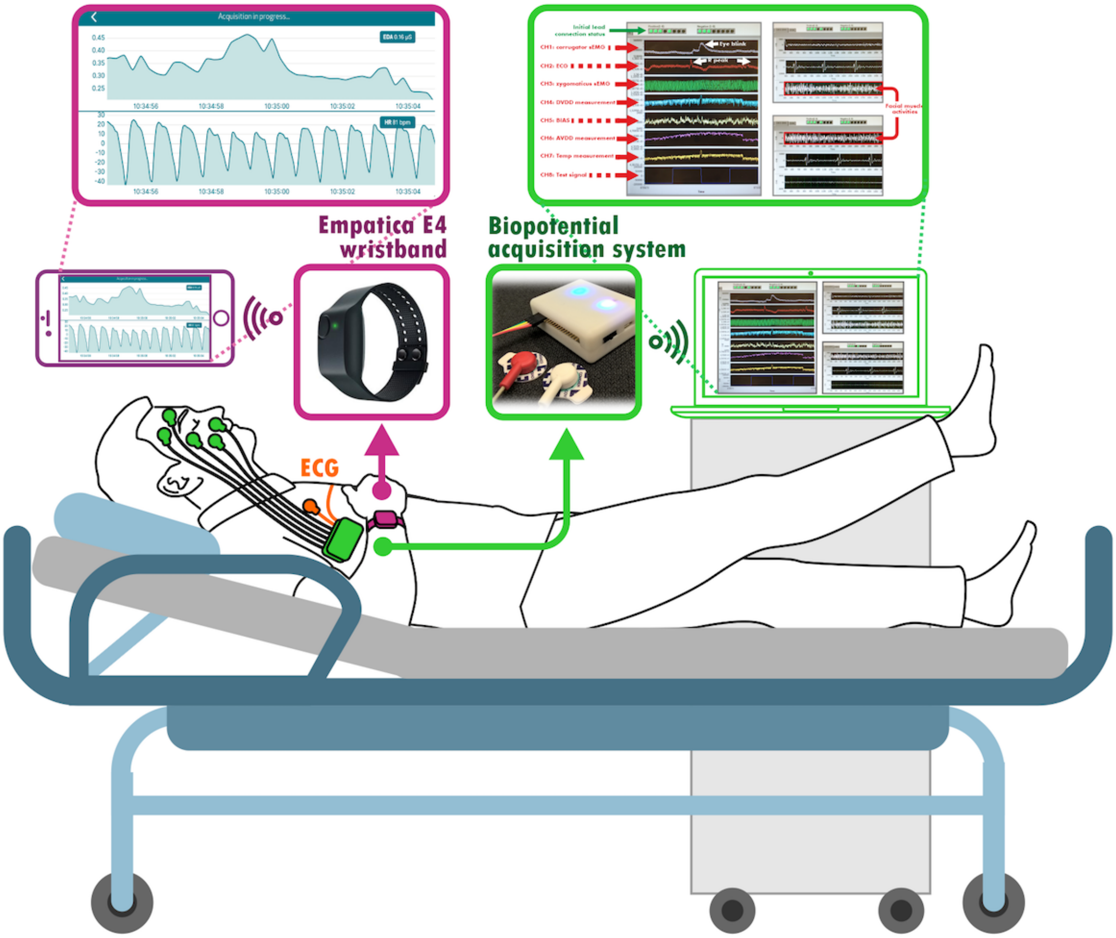


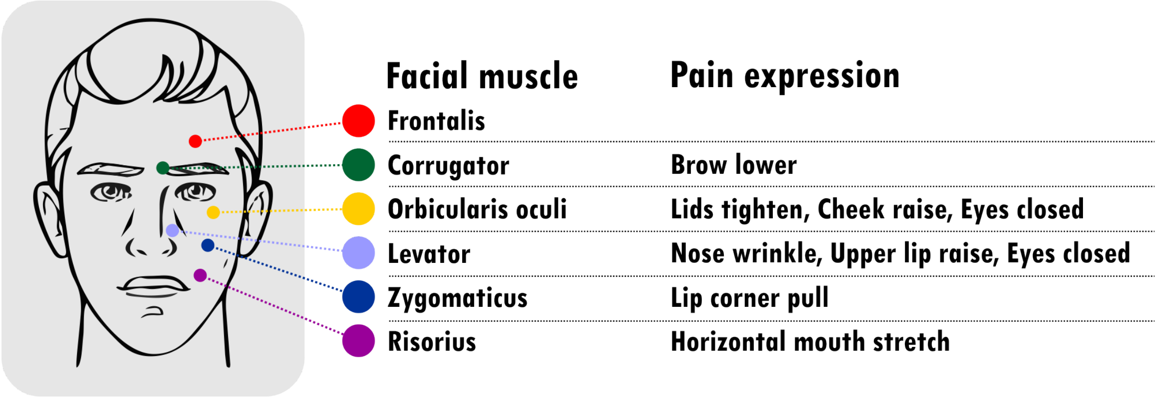
Figure 3.1: Setup of the biopotential acquisition system on the patient

trodes, electrode-to-device lead wires, an ADS1299-based portable device, and computer software (LabVIEW version 14.02f, National Instruments) to visualize data streaming from the portable device. Raw signals from the electrodes are sampled at 500 samples per second and are sent to the computer software via Bluetooth for visualization [153].

Facial sEMG and ECG signals were sampled at 500 Hz. The ADC resolution of ADS1299 is 24-bit. Its analog input range was between -4.5 V/gain and 4.5V/gain as the internal reference 4.5 V was used and the gain was set to be 24 in the data collection. Therefore, the input voltage range was between -187.5 mV and 187.5 mV. The laptop powered the data acquisition hardware through a USB cable which was also the data transmission channel.

The laptop ran on battery power during the data collection to avoid bringing additional powerline interference to sEMG signals.

Facial sEMG: The facial muscle areas were chosen based on Figure 3.2. The single electrode for each area was placed on the left side of the face following the facial EMG electrode placement guidelines [48]. Muscle frontalis is not involved in pain facial expressions in existing literature. Its signal was taken as a noise reference to all the other sEMG signals. Before attaching the H124SG electrodes to the face, the electrode sites were wiped with an alcohol pad as skin preparation.



Facial muscle	Pain expression
● Frontalis	
● Corrugator	Brow lower
● Orbicularis oculi	Lids tighten, Cheek raise, Eyes closed
● Levator	Nose wrinkle, Upper lip raise, Eyes closed
● Zygomaticus	Lip corner pull
● Risorius	Horizontal mouth stretch

Figure 3.2: Facial muscles reflecting pain

ECG: Two lead ECG electrodes were placed on left and right arm. ECG waveforms were collected with the same device as sEMG signals. The signals from left arm and right arm were differentiated to neglect the effect of powerline interference to ECG signals and reduce the baseline wandering noise. The two sensor pads were moistened before use to aid conductivity, and the strap should be adjusted to be a snug fit.

Empatica E4

We use the commercially available Empatica E4 wristband (Empatica Inc, Boston, MA, USA) [45] to measure EDA and PPG signals. The wristband is simple to position, and

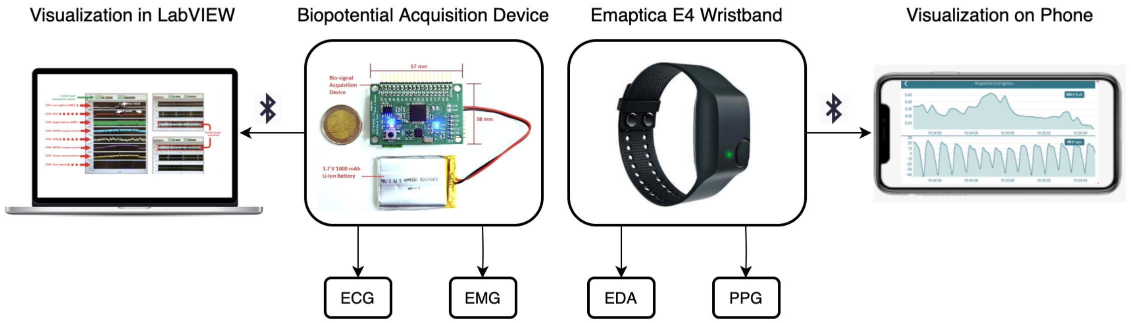

Visualization in LabVIEW
Biopotential Acquisition Device
Emaptica E4 Wristband
Visualization on Phone
57 mm
58 mm
Bio-signal Acquisition Device
3.7 V 1000 mAh Li-ion Battery
ECG
EMG
EDA
PPG

Figure 3.3: Biosignal Acquisition using the iHurt System. The signals collected were electrocardiogram (ECG), electromyogram (EMG), electrodermal activity (EDA), and photoplethysmogram (PPG).

participants can maneuver easily without the device impeding their movements in any way. The wristband's internal memory allows recording up to 36 hours of data and wireless data transmission. The E4 wristband is rechargeable, with a charging time of fewer than 2 hours. The E4 was connected to the participants' phone over Bluetooth for visualization. Figure 3.3 presents a visual depiction of our system.

EDA: An EDA sensor is embedded in the E4 wristband. This sensor measures the fluctuating changes in certain electrical properties of the skin at a 4Hz sampling rate.

PPG: Same wristband (E4) was used to collect PPG data with a 64 Hz sampling rate. This sensor measures the variation of blood flow by emitting a light onto the surface of the skin and measuring the light absorption. The PPG signal consists of a pulsatile (AC) component and a non pulsatile (DC) component [3]. The AC component reflects the pulsations in the interrogated blood volume with each heartbeat, whereas the DC component contains the low frequency fluctuations, including absorption from the tissue and bones as well as static blood absorption [130]. The AC component –oscillated with the contraction and relaxation of the heart– enables the measurements of cardiac cycles by detecting the peaks (i.e., maximum values) in this signal. We used an Empirical Mode Decomposition (EMD) based method proposed by Madhav et al. [103] to derive respiration signals from PPG.

Chapter 4

Signal Quality Monitoring

Continuous monitoring of patients using wearable sensors has changed the landscape of healthcare, providing solutions to many of the open challenges of smart affective computing applications. Signal Quality Monitoring handles quality assurance and improving the reliability of physiological measurements obtained from signals recorded via wearable sensors which are more prone to artifacts. Any disruptive event in input data of a specific modality (e.g., motion artifacts or sensor detachment) needs to be addressed appropriately. This chapter explains the Signal Quality Monitoring block shown in Figure 1.1 in detail and makes the case on how important is signal quality monitoring and proposes a deep learning-based PPG quality assessment method for HR and various HRV parameters as PPG is fast becoming the most popular monitoring tool because of its ease of measurement. We employed one customized 1D and three 2D Convolutional Neural Networks (CNN) to train models for each parameter. Reliability of each of these parameters is evaluated against the corresponding electrocardiogram signal, using 210 hours of data collected from a home-based health monitoring application.

4.1 Introduction

Internet of Things (IoT) technology is fundamentally changing the delivery of healthcare, enabling health monitoring applications anywhere and anytime [59, 109, 16, 15, 13]. Wearables, as intelligent electronic devices, play a key role in such applications; health data are collected, analyzed, and shared over a network. Such devices are becoming widely used around the globe, as they are even more miniaturized, smarter, and easier to use. Wearables can include a variety of sensing resources to continuously monitor body functions. Photoplethysmography (PPG) is an inexpensive and convenient method that is being employed in a variety of wearables as well as smartphones to collect various vital signs such as heart rate (HR), heart rate variability (HRV), SpO2, and respiration rate [3].

PPG is a simple optical method to measure plethysmogram, showing the variations of blood volume in body organs. The obtained signal can be tailored to track cardiorespiratory parameters. The PPG method mainly consists of two components placed on the skin. First, a light source is utilized to reflect light to the skin surface. The red, infrared, or green light can be selected according to the application. Second, a photodetector collects the light reflection [176]. The collected signal includes a pulsatile (AC) and a slowly fluctuating (DC) component, allowing the monitoring of cardiorespiratory parameters non-invasively and continuously. The PPG method is tailored in many clinically approved devices and commercial wearables (e.g., smartwatches and rings), as it is easy-to-implement and energy-efficient [105, 119, 38, 6, 5, 9].

On the other hand, input PPG signals might be distorted due to noises caused by motion artifacts and other environmental sources [102], which are ubiquitous and unavoidable in everyday life settings. For instance, the light sensors might be exposed by environment light sources, by which the collected PPG signal is distorted and the information is concealed within the signal. Specifically, we observe noise affects the collected signal differently as the users participate in various physical activities while using the PPG-based wearables. Such movements could negatively impact the signal quality [203, 64]. For example, if the engaged subject is running, the noise power is much higher compared to that affecting the same signal acquired when the participant is sleeping. Low-quality PPG signal (i.e., low signal-to-noise ratio (SNR)) affects the reliability of the health parameters extracted: e.g., HR and HRV parameters. Such unreliable measurements can lead to false alarms or life-threatening decisions in healthcare applications.

In the literature, the PPG signal quality was investigated by proposing assessment methods to discriminate reliable and unreliable parts of the signal. Various studies introduced signal quality indicators [44, 169, 159] or utilized template matching approaches to distinguish the reliable segments of the PPG signal [174, 120, 72]. Moreover, traditional machine learning methods such as support vector machine (SVM), decision tree, and K-nearest neighbors were presented to carry out PPG quality assessment using features extracted from the shape of the signal [2, 201, 127, 149, 128, 96, 104]. Recently, deep learning methods were also exploited PPG quality assessment, enabling automatic PPG feature extraction [126, 170, 111, 54, 145].

These studies have mostly assessed the shape of the signal, focusing on the HR. In other words, the signal is classified as reliable if the cardiac cycle can be detected. We believe that such a quality assessment method is insufficient for PPG signals, from which many other parameters can also be extracted. PPG signals can be leveraged to remotely collect HRV parameters whose accuracy might be diminished due to different factors in the signal. A PPG signal might be reliable for HR detection but, for example, unreliable for standard deviation of NN intervals (SDNN). Therefore, a PPG quality assessment method is essential in health monitoring applications, determining the reliability of the PPG signal according to the health parameters. Such a method needs to provide a confidence value for each health parameter. Consequently, unreliable parameters can be removed, preventing incorrect health decision-making.

Moreover, there are available public datasets, such as WESAD, including physiological and motion data recorded using wrist- and chest-worn devices in lab settings [155]. PPG signals are often distorted due to motion artifacts and environmental noise. Therefore, data collected in lab settings are insufficient for PPG signal quality assessment studies. The signal quality assessment methods should be evaluated using the PPG data collected in free-living conditions, where the users engage in their daily routines.

We propose a quality assessment method to distinguish reliable PPG signals according to the HR and HRV values extracted from the signal. Convolutional Neural Network (CNN) methods are tailored in this regard to train models for each health parameter. We design i) a customized 1D CNN method and ii) three 2D CNN methods enabled by three deep neural networks. The proposed methods are investigated, and the best architecture is selected. Then, the performance of the selected method is evaluated, in comparison to existing rule-based, machine learning, and deep learning PPG quality assessment methods. To this end, we perform a home-based Electrocardiogram (ECG) and PPG collection, in which the signals are acquired simultaneously and remotely for 24 hours. The evaluation includes more than 210 hours of data. For each health parameter, the PPG quality is defined in comparison to the ECG using an automatic annotation process.

4.2 Related Works and Motivation

Several signal quality assessment techniques have been introduced in the literature to determine whether collected PPG signals are reliable. To this end, the unreliable part of the signal is detected and removed, preventing misinterpretation of data and invalid decision-making. Such techniques are mostly focused on the morphology of the bio-signals. Different signal quality indicators –such as skewness, kurtosis, and baseline wandering of the signal– were exploited to estimate the PPG signal quality [44, 169, 159]. Rule-based methods have also been introduced to distinguish low-quality bio-signals, leveraging decision rules: e.g., signal saturation detection and beat-to-beat-interval evaluation [115, 144, 186]. Moreover, template matching techniques have been proposed to distinguish reliable and unreliable signals. For example, Sun *et al.* [174] proposed a template matching method based on dynamic time warping for signal quality assessment. In other similar studies, the quality assessment was performed by investigating the morphological similarity between the original signal and the template signals, which were the expected waveform or surrounding pulses [120, 72]. The PPG morphological waveform can be affected by motion artifacts, environmental noise, and cardiovascular issues. Therefore, these methods are inaccurate due to variations in the PPG morphological features. Moreover, these methods usually need predefined thresholds that should set manually based on the data. A number of machine learning methods have been developed for PPG signal quality assessment. Various studies in the literature proposed to train machine learning methods, utilizing morphological and time/frequency domain features of PPG signals. These methods utilized supervised and lightweight unsupervised approaches. The supervised methods include hierarchical decision rule [187], decision tree [95], random forest [2], support vector machine (SVM) [201, 127], and neural network [96]. With this intention, Sabeti *et al.* [149] proposed a threshold optimization learning method and compared the method with a SVM and a decision tree. Furthermore, in [128], the authors investigated SVM, K-nearest neighbors, and decision tree for the signal quality assessment in the case of atrial fibrillation. In addition, Mahmoudzadeh *et al.* [104] proposed a lightweight unsupervised method providing low-computational real-time PPG quality assessment. Authors in [148] also proposed unsupervised PPG SQA methods based on the self-organizing map. Machine learning-based methods outperform rule-based methods in terms of accuracy. However, similar to the rule-based method, results are affected by morphological variation of PPG signals. In addition, machine learning-based methods utilized manual feature extraction. Therefore, the accuracy and generalization of these methods are restricted due to a limited set of selected features.

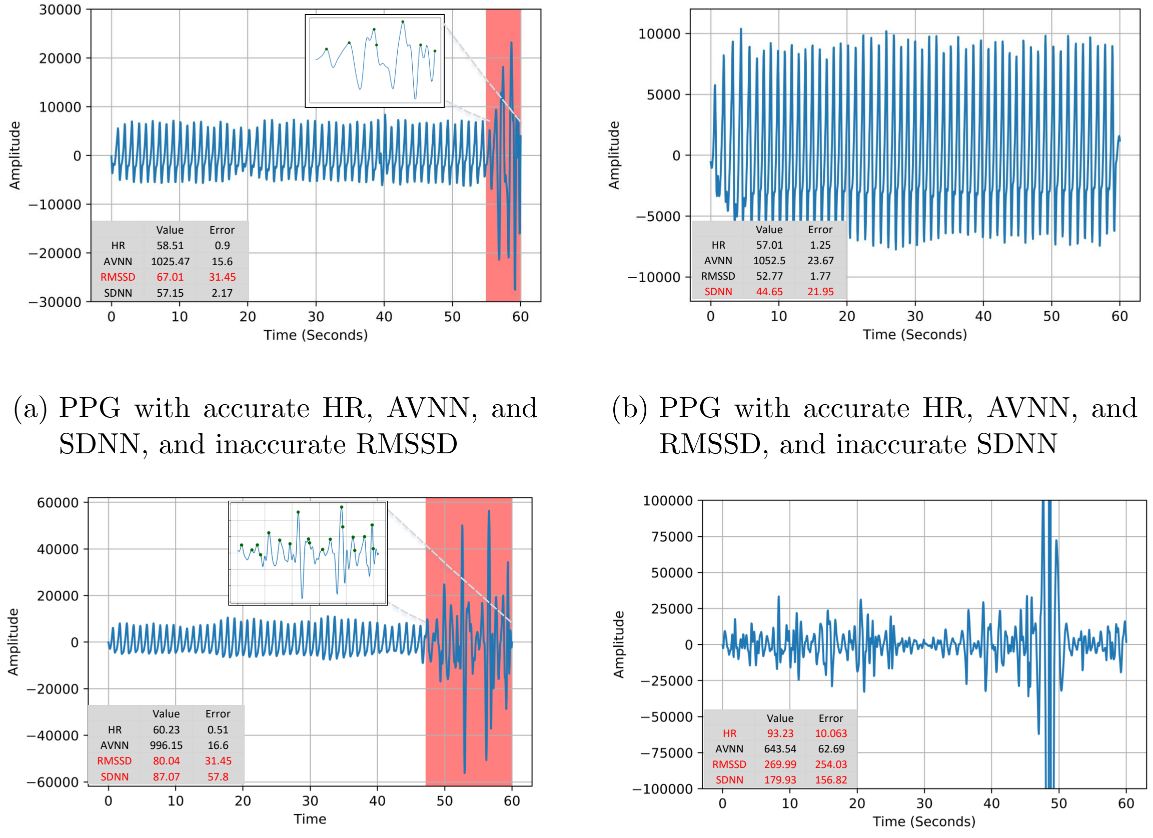
Recently, deep learning methods were also introduced for the PPG quality assessment. Earlier, we (Kasaeyan Naeini *et al.*) [111] introduced a real-time PPG assessment approach to classify the signals according to the HR values. Perieira *et al.* [126] used 1-dimensional (1D) and 2-dimensional (2D) deep neural networks for signal quality assessment in case of atrial fibrillation. They considered the PPG signal as time series (1D) and also as images (2D). Authors in [54] also proposed a 1D CNN model for determining reliable segment of PPG signals. In another work, Soto *et al.* [170] proposed a multi-task CNN called DeepBeat for signal quality assessment and atrial fibrillation detection. They improved their results by pre-training the model using convolutional denoising auto-encoders (CDAE). In addition, Roh *et al.* [145] utilized a 2D CNN for classifying each segment of beat waveform. These methods benefited from automatic feature extraction. However, they need manual/expert labeling based on HR values or morphological waveform of PPG signals. They are inappropriate for HRV parameters analysis. A comparison of the PPG quality assessment methods is shown in Table 4.1.

The PPG quality assessment methods, presented in the literature, were mostly designed to analyze the shape of a signal and assessing the reliability of the PPG signal itself focusing solely on HR. These methods tailor rules, templates, and hypothesis functions to classify a PPG as reliable if the desired signal (i.e., cardiac cycle) is appropriately retrieved. Such methods can be utilized for HR detection applications where the pulse is visible in the reliable signals. However, they are not applicable to HRV analysis, as only having an acceptable shape is insufficient, and other factors in the signal may affect the accuracy too. Moreover, such factors could impact differently on the accuracy of the HRV parameters. For instance, a five-minute high-quality PPG signal with two or three noisy peaks could result in an accurate/acceptable HR and SDNN, but can lead to an invalid RMSSD value.

To emphasize the importance of developing quality assessment models for each HRV parameter, and for clarification, let us show five real motivational examples of PPG signals

	Reference	Features	Method	Annotation	Source code Availability
	Proposed method	Automatic feature extraction	1D CNN/ 2D CNN	Automatic labeling based on HR and HRV parameters	✓
Rule based Methods	Orphanidou et al [115]	Extracted HR and morphological features	Threshold based rules	Manual based on the signal shape	×
	Reddy et al [144]	Predictor Coefficient	Hierarchical decision rules	”	×
	Tyapochkin et al [186]	Statistical parameters of IBIs	Predefined rules	”	×
	Vadrevu et al [187]	Amplitude and time-series features	Hierarchical decision rules	”	×
Supervised methods	Alam et al [2]	Morphological features and HR value	Random forest	Manual based on the signal shape and HR values	×
	Zhang et al [201]	Frequency domain and time series characteristics	SVM	Manual based on the signal shape	×
	Preira et al [127]	Frequency-domain, time-domain, and non-linear features	SVM	”	×
	Preira et al [128]	Frequency-domain, and time-domain features	SVM	”	×
Unsupervised methods	Mahmoudzadeh et al [104]	Statistical features	Elliptical envelope	”	✓
	Roy et al [148]	Entropy and signal complexity features	Self-organizing map	”	×
Deep learning methods	Preira et al [126]	Automatic feature extraction	ResNet18	”	×
	Soto et al [170]	”	Multi-task CNN (pre-training with CDAE)	”	×
	Goh et al [54]	”	1D CNN	”	×
	Roh et al [145]	”	2D CNN	”	×
	Naeini et al [77]	”	1D CNN	Automatic labeling based on HR	×

Table 4.1: Comparing PPG quality assessment methods



	Value	Error
HR	58.51	0.9
AVNN	1025.47	15.6
RMSSD	67.01	31.45
SDNN	57.15	2.17

(a) PPG with accurate HR, AVNN, and SDNN, and inaccurate RMSSD

	Value	Error
HR	57.01	1.25
AVNN	1052.5	23.67
RMSSD	52.77	1.77
SDNN	44.65	21.95

(b) PPG with accurate HR, AVNN, and RMSSD, and inaccurate SDNN

	Value	Error
HR	60.23	0.51
AVNN	996.15	16.6
RMSSD	80.04	31.45
SDNN	87.07	57.8

(c) PPG with accurate HR and AVNN, and inaccurate RMSSD and SDNN

	Value	Error
HR	93.23	10.063
AVNN	643.54	62.69
RMSSD	269.99	254.03
SDNN	179.93	156.82

(d) PPG with (relatively) accurate AVNN, and inaccurate HR, RMSSD, and SDNN

Figure 4.1: One-minute windows of filtered PPG signals, from which HRV parameters with different accuracy are extracted

with different artifacts in Figures 4.1 and 4.2. These figures show that a noisy PPG signal, which results in some unreliable HRV parameters, can still be used to extract other HRV parameters accurately. Therefore, the quality of PPG signals should be evaluated according to the desired HRV parameters. In these examples, we extract the HR, AVNN, RMSSD, SDNN, and LF/HF ratio of the PPG signals. Then, we specify their errors by calculating the distance between these parameters and the corresponding parameters obtained from the baseline ECG signals. In other words, the error is $ECG\ value\ -\ PPG\ value$.

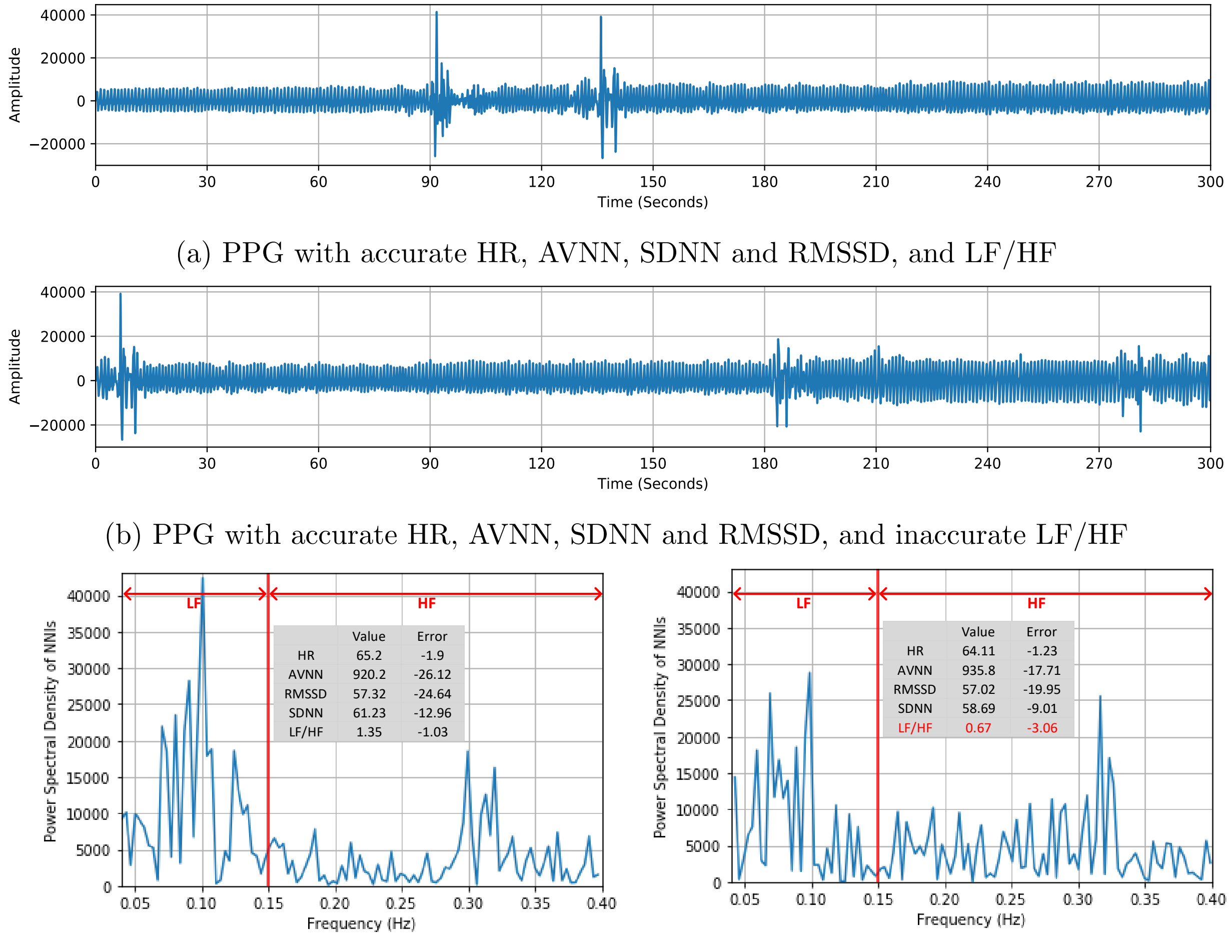
a) Figure 4.1a illustrates an example, in which less than 5 seconds of the PPG signal (highlighted in red) is distorted due to hand movements. As indicated, a few peaks are detected incorrectly in the noisy part. The PPG signal provides HR, AVNN, and SDNN values with low errors compared with the ECG. However, the error for the RMSSD is high. The reason is that RMSSD is correlated to the short-term variation in the PPG signal, and a small corrupted part in the signal could affect its accuracy.

b) Figure 4.1b shows a PPG signal with no distorted peaks. The cardiac cycles can be extracted appropriately. However, the variation of the intervals is not similar to the ECG. This signal provides reliable HR, AVNN, and RMSSD but unreliable SDNN. SDNN shows the long-term variation in the signal. Therefore, noises that affect the variation of the signal will negatively impact its accuracy.

c) Figure 4.1c shows a one-minute window where the PPG signal is partially corrupted (i.e., 20%). As more peaks are detected incorrectly due to the noise, both short-term and long-term variations in the interval distribution will be affected. Therefore, both RMSSD and SDNN are unreliable in this signal although the extracted HR and AVNN are acceptable.

d) Figure 4.1d depicts a noisy PPG signal, where extracted HR, RMSSD, and SDNN are inaccurate. Although a considerable amount of the signal is corrupted, the error of the AVNN is relatively low. Since AVNN represents the mean of the intervals, it is more resistant to outliers and noises. This example shows that we can extract some HRV parameters with acceptable accuracy although the signal is noisy.

e) Figures 4.2a and 4.2b show two five-minute samples of PPG signal with accurate HR, AVNN, SDNN, and RMSSD. However, the signal indicated in Figure 4.2a is reliable for extracting LF/HF, while the signal in Figure 4.2b is unreliable for the same feature. Figures 4.2c and 4.2d show power spectral density (PSD) of NNIs for these two samples in Figure 4.2a and Figure 4.2b, respectively. As indicated, although the two PPG signals (in the time domain) are similar, the power of the NNIs signals in the low



	Value	Error
HR	65.2	-1.9
AVNN	920.2	-26.12
RMSSD	57.32	-24.64
SDNN	61.23	-12.96
LF/HF	1.35	-1.03

	Value	Error
HR	64.11	-1.23
AVNN	935.8	-17.71
RMSSD	57.02	-19.95
SDNN	58.69	-9.01
LF/HF	0.67	-3.06

(a) PPG with accurate HR, AVNN, SDNN and RMSSD, and LF/HF

(b) PPG with accurate HR, AVNN, SDNN and RMSSD, and inaccurate LF/HF

(c) PSD of NNIs of the PPG signal in Figure 4.2a

(d) PSD of NNIs of the PPG signal in Figure 4.2b

Figure 4.2: Two filtered PPG samples with corresponding PSD of NNIs

frequency and high frequency bands are different. The reason is that the LF/HF is a frequency domain feature and will be distorted if noise with the same frequency of low frequency or high frequency band is added to the signal.

These examples show that HRV parameters are of essence for PPG quality assessment and those PPG quality assessment methods which are solely based on HR or the morphology of the PPG itself are not efficient and can result in many adverse consequences, some of which may be life-threatening due to not fully correct decisions made by doctors. Moreover, state-of-the-art PPG quality assessment methods were mostly evaluated using the simulated data or data collected in controlled lab settings with limited motion artifacts. We believe that the confidence models need to be trained using the data collected in everyday settings where the subjects engage in several physical activities in various environments. This way, the model can learn about the validity of the signal in different conditions with different artifacts.

4.3 Background

Recent advances in information and communication technology (ICT) provide an opportunity to enable remote health monitoring using wearable electronics. Such wearables can measure biomedical signals, allowing continuous monitoring of the individual's health condition. ECG is a non-invasive method which can be used to remotely track cardiorespiratory parameters using portable monitors [105, 138]. The method includes limb and chest electrodes, which collect electrical signals generated from the action potentials of the heart cells. The ECG is the gold standard in HR detection and diagnosis of cardiovascular diseases. However, it cannot be performed for long-term monitoring due to its complicated data collection. Alternatively, PPG is a more convenient method to monitor cardiorespiratory variables. The PPG acquires the rate of blood flow in the tissue (e.g., wrist) as controlled by the heart's pumping action. The method leverages an optical sensor in conjunction with a light source to collect the signals. In the following, we outline background on the PPG, HRV, and Convolutional Neural Networks (CNNs) as a method we used for the PPG analysis.

4.3.1 Photoplethysmography

PPG is an optical measurement method which records the variation of blood flow by emitting a light onto the surface of the skin and measuring the light absorption. The PPG signal

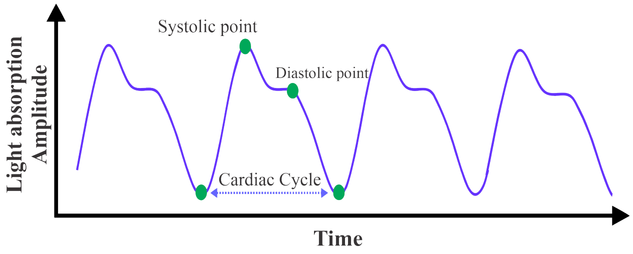


Figure 4.3: A window of a filtered PPG waveform

consists of a pulsatile (AC) component and a non pulsatile (DC) component [3]. The AC component reflects the pulsations in the interrogated blood volume with each heartbeat, whereas the DC component contains the low frequency fluctuations, including absorption from the tissue and bones as well as static blood absorption [130]. The AC component –oscillated with the contraction and relaxation of the heart– enables the measurements of cardiac cycles by detecting the peaks (i.e., maximum values) in this signal. The PPG signal calculated with this procedure can provide the real-time measurements of HR. Moreover, variation in time intervals between the successive pulse peaks, called Heart Rate Variability (HRV) allow us to obtain more information about the Autonomic Nervous System (ANS). Figure 4.3 shows an individual PPG signal.

The PPG method is convenient, economic, and easy-to-set up [34] using an optical sensor in conjunction with a light source such as a green LED. The method is already used in various commercial and clinical devices. The PPG with green light is utilized in the optical sensors of most consumer wearable devices such as smartwatches for HR calculation. In addition, the PPG with red and infrared LEDs are employed in pulse oximeters to monitor peripheral capillary oxygen saturation (SpO_2).

4.3.2 HRV analysis

HRV consists of the fluctuations in the time periods between successive heartbeats [107]. In the literature, HRV analysis has been introduced to examine the Autonomic Nervous System

(ANS) correlated with pain intensity, stress level, sleep quality, to name but a few [71, 77, 150, 42]. Conventionally, HRV values are calculated from the ECG signal by extracting the cardiac cycles: i.e., the RR-intervals in the signal. Alternatively, the HRV can be also obtained using the PPG signals, where the peak-to-peak intervals –also called NN intervals (NNIs)– indicate the cardiac cycles. Studies show that there is a high correlation between the HRV obtained from the ECG and PPG [179, 157, 23]. The HRV values are extracted over a period of the ECG/PPG signal, which is in long-term over 24 hours, in short-term over 5 minutes, or in ultra-short-term over 1 minute [160, 150, 65].

The HRV can be obtained from the signal both in time domain and frequency domain. The time domain features of HRV are statistical features that quantify the amount of variability in measurements of the inter-beat-interval (IBI), which is a time interval between adjacent heartbeats. In contrast, the frequency domain features of HRV are estimations of distribution of absolute or relative power into four frequency bands mainly based on Power Spectral Density (PSD). Heart rate oscillations is divided into ultra-low-frequency (ULF), very low-frequency (VLF), low-frequency (LF), and high-frequency (HF) bands Force [29]. Some common HRV features in both time domain and frequency domain are described in Table 4.2.

The HRV features obtained from the PPG show different characteristics within different sampling frequencies. The lower the sampling frequency is, the more variation in the peak locations, meaning the more errors in the HRV analysis [35]. Considering this fact and limitation of our PPG data collection (the sampling rate was 20Hz), we only focus on the HR and the following four HRV features. These features, which show insignificant error rate at $f_s \geq 20$, are AVNN (Average of NN intervals), RMSSD (Root Mean Square of Successive Differences), SDNN (Standard Deviation of NN intervals) from time domain and LF/HF (ratio of LF power and HF power) from the frequency domain [35]. We select these HRV in this study, as they are important for various health application such as stress

Table 4.2: Time domain and Frequency domain HRV features and the descriptions

Feature	**Units**	**Description**
AVNN	ms	Mean of NN intervals
SDNN	ms	Standard deviation of NN intervals
RMSSD	ms	Root mean square of successive NN interval differences
SDSD	ms	Standard deviation of successive NN interval differences
nnXX	ms	Number of NN interval differences greater than the specified threshold
pnnXX	%	Percentage of successive NN intervals that differ by more than x ms
VLF power	s^2	Absolute power in very low frequency band (≤ 0.04)
LF power	s^2	Absolute power in low frequency band ($0.04 - 0.15$)
HF power	s^2	Absolute power in high frequency band ($0.15 - 0.4$)
LF peak	Hz	Peak frequency in low frequency band ($0.04 - 0.15$)
HF peak	Hz	Peak frequency in high frequency band ($0.15 - 0.4$)
Total Power	s^2	Total power over all frequency bands
LF/HF	%	Ratio of LF-to-HF power

monitoring [150].

4.3.3 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are a class of deep neural networks, also known as the state-of-the-art models for image recognition problems [92]. CNNs are capable of automatically learning spatial hierarchies of features, from low- to high-level patterns. CNN is a hierarchical model consists of convolutional, subsampling, and fully connected layers. The first two layers –convolution and subsampling– carry out the feature engineering part and the third layer –a fully connected layer– performs the classification.

The state-of-the-art deep architectures such as VGG, ResNet, MobileNet implement different approaches for the classification tasks. The VGG was proposed to increase the depth of the convolutional structure of the model for achieving better performance [166]. VGG obtained a top-5 error rate of 7.32%. However, only increasing the depth of the network, saturates the accuracy and then degrades it rapidly. Therefore, the ResNet was introduced to address this problem exploiting the shortcuts or parallel blocks of convolutional filters while building deeper models [66]. ResNet, however, achieved a top-5 error rate of 3.57%. In contrast, MobileNet was proposed as a light weight model to run deep neural networks on personal mobile devices. MobileNet obtained a top-5 error of 7.5%, almost the same as VGG Network. We leverage the significant performance of these deep architectures for the PPG quality assessment. In this regard, we convert the PPG signals to PPG snapshots to feed the images to the CNNs.

Furthermore, CNNs can be harnessed on one-dimensional time-series data sharing the same characteristics and the same approach as in image-based CNNs [49]. The difference is the structure of the input data and how the filter –i.e., convolution kernel or feature detector– slides across the data. The model learns to extract features using a technique called sliding window with a one-dimensional filter over the time-series, followed by a non-linear function to learn non-linear decision boundaries. The model can learn an internal representation of the time-series data automatically.

4.4 Methods and Materials

The data used in this work is part of a multipurpose study on remote health monitoring. In the following, we briefly describe the participants, recruitment, data collection, and data annotation in this study.

4.4.1 Participants and Recruitment

The study was conducted in southern Finland during July-August 2019 by inviting healthy individuals who were between 18 and 55 years old. The exclusion criteria were if the possible candidates had a diagnosed cardiovascular disease, symptoms of illness during the recruitment, and restrictions on the physical activity or using the devices in the daily routines. The recruitment started by personally contacting students and staff members of the University of Turku. Then snowball sampling was used to reach a convenient number of participants. We aimed for equal number of female and male participants as gender affects HRV parameters. In face-to-face meetings, the selected candidates were informed about the purpose of the study and the instructions to use the devices: i.e., a Shimmer device [164] and a Samsung Gear Sport smartwatch [151]. Written informed consent forms were also provided to the participants. Forty-six individuals, who agreed to participate in this study, were asked to wear the devices for 24 hours continuously.

4.4.2 Ethics

The study was conducted according to the ethical principles based on the Declaration of Helsinki and the Finnish Medical Research Act (No 488/1999). The study protocol received a favorable statement from the ethics committee (Unversity of Turku, Ethics committee for Human Sciences, Statement no: 44/2019). The participants were informed about the study both orally and in writing, before their consent was obtained. Participation was voluntary and all the participants had the right to withdraw from the study at any time and without giving any reason. To compensate the time used for the study, each participant got a gift card to grocery store (20 euro) at the end of the monitoring period when returning the devices.

4.4.3 Data Collection

We performed home-based ECG and PPG collection, in which the signals were acquired simultaneously and remotely for 24 hours. In this study, the ECG signal was collected, employing the Shimmer3 ECG [164]. The ECG test included 4 limb leads placed on the left arm, right arm, left leg, and right leg. This device provides raw data using medical grade

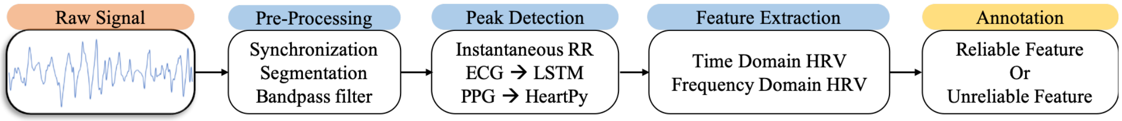


Figure 4.4: Automatic PPG Annotation Pipeline

sensors. It also provides accelerometer, gyroscope, and magnetometer data. Participants were instructed to place the ECG unit on their chest and to attach the four skin electrodes. The 512Hz sampling rate was used in this study, as suggested for clinical trials.

In addition, the Samsung Gear Sport smartwatch was selected for collecting the PPG signal, considering availability of the raw PPG signals and configurability of the data recording. The watch also has a built-in inertial measurement unit (IMU) by which daily physical activity data are extracted. Participants were asked to wear the watch on the non-dominant hand continuously and tightly enough. Considering constraints of the battery in the Gear sport watch, we programmed the watch to collect 16 minutes PPG signal in every 30 minutes. In this settings, there is no need to charge the smartwatch during the one-day experiment. Each PPG record contains one minute of unreliable data (due to sensor calibration) and 15 minutes PPG signals. The PPG signals were collected with 20Hz sampling frequency that is suitable to extract HR and HRV parameters.

4.4.4 Automatic PPG Annotation

The PPG signals should be annotated to develop a quality assessment method. Traditionally, the signals are manually annotated, where experts label windows of the signals into "reliable" or "unreliable" according to the shape/structure of the signals. As described in Section 4.2, such a method is inaccurate when multiple parameters are obtained using the PPG. The PPG signals should be labeled according to the accuracy of the desired applications/health parameters. Therefore, different labels should be allocated to a window of the signal: e.g., the window is reliable for HR and SDNN while it is unreliable for RMSSD (see Figure 4.1

and Figure 4.2).

To address this issue, we develop an automatic PPG annotation method, where the PPG signals are labeled according to the health parameters. In this regard, the obtained parameters are compared with the parameters extracted from the ECG as the baseline method. The signal is labeled as “reliable” for a parameter if the error is insignificant. Otherwise, it is labeled as “unreliable”. A schematic representation of the automatic PPG annotation method is shown in Figure 4.4.

Pre-Processing

The ECG and PPG signals were collected by two different wearables for 24 hours. Therefore, there might be a time shift between the two signals (e.g., seconds). To address this issue, we first synchronize the two signals. We used a cross-correlation method to synchronize the data provided by the ECG device and the smartwatch. The ECG device and PPG-based smartwatch collected acceleration signals with the same frequency of ECG and PPG signals, respectively. We extracted the cross-correlation of the signal vector magnitudes of the acceleration signals. The output indicated the possible time shift between the two signals. Then, the ECG signals was shifted with respect to the PPG signals if needed. We then segment the 24-hour PPG data into five-minute windows using sliding technique with an stride of 10-seconds. Considering HR between 30 to 220 beats per minute, a Butterworth filter [156] was set to only pass heartbeat signals (i.e., 0.5-3.7 Hz).

Peak Detection

The first step of the HRV analysis is to calculate the RR intervals. In the ECG analysis, the RR intervals are calculated by detecting the QRS complex –with the highest peak and slope in the signal– and extracting the distance between two adjacent R peaks. For the QRS detection, we used the method proposed by Laitala et al [90], as it shows more robust and accurate QRS detection in comparison to the traditional QRS detection methods such as Pan-Tompkins [118], Christov [37], Hamilton [62], and Engzee [46, 99]. This method uses a Long Short Term Memory (LSTM) network to obtain the probabilities and locations of the peaks, followed by extra processes to remove outliers based on anomalous peak-peak distances and obtain the valid peaks. The last step is to remove noise from the RR intervals –the time intervals between two successive R-peaks– obtained by subtracting the time of two successive peaks. We used a quotient filter [131] to remove outliers from the RR intervals that are used for the feature extraction.

In the PPG analysis, HRV parameters are obtained from the the subtle change of pulse periods –i.e., inter beat intervals (IBI)– generated as a result of the heart activity. To extract the IBI from the PPG signals, we use a peak detection method proposed by Van Gent et al [190]. They showed the method obtains an acceptable accuracy to extract HRV values in comparison to reference ECG signals using Pan-Tompkins QRS algorithm [118] and an open source algorithm called HRVAS ECGViewer [141]. This method uses an adaptive threshold to accommodate morphology variation in the PPG waveform, followed by an outlier detection/rejection to extract valid peaks in the signal.

Feature Extraction

The HRV parameters can be calculated using the extracted peaks. In this study, we only extract 4 HRV parameters, as the PPG sampling frequency is 20Hz [35] (see Section 4.3.2). Within each time window of the both ECG and PPG, HR and the 4 HRV features –three time domain and one frequency domain– are extracted. The time domain metrics are obtained using the NN intervals: **AVNN** is the Average Value of NN intervals, **RMSSD** is the Root Mean Square of Successive Differences between normal heartbeats, and **SDNN** is the Standard Deviation of NN intervals. **LF/HF** is also the ratio of the low-frequency to high-frequency power. In our setup, these features are extracted from the ECG and PPG signals using the HeartPy Python package [190].

Labeling

As previously mentioned, the PPG signals are divided into five-minute windows, from which five features – HR, AVNN, RMSSD, SDNN, and LF/HF – are extracted. The features are tailored to label the PPG windows as "reliable" or "unreliable." For each window, the extracted features are compared with the values obtained from the corresponding ECG signal using an Euclidean distance function. The window is labeled as "reliable" for a feature if the distance is less than a threshold value obtained according to the range of the feature [125]. Otherwise, the window is labeled as "unreliable." Noted that the threshold can be selected according to the desired accuracy of the feature. This process is performed for the five features. Therefore, five binary labels are allocated to each PPG window.

4.5 PPG Quality Assessment Approach

In this section, we present two deep learning (DL) based methods using CNNs for PPG Quality Assessment. CNN architectures are capable of handling the challenging feature engineering of PPG signals automatically. This is an advantage over the traditional PPG Quality Assessment methods which were mostly designed to extract features based on the morphology of the PPG itself. The traditional methods use extracted features to generally classify the PPG signal as "reliable" or "unreliable" solely based on HR, however, we create a specific model for each of the HR and HRV parameters separately. The classification is performed using the labels generated from our automatic ECG-based annotation method. In addition, our data include more scenarios and corner cases with different artifacts compared to a lab setting based data collection, as our data were collected in the course of everyday events.

CNN can perform feature extraction and classification without having any knowledge about the data collection. Moreover, CNN architectures naturally can handle an input with any dimensionality; two dimensional and one dimensional inputs are the most common ones. PPG signals as a time-series have potential to be converted to an image if a 2D model is desired. With this privilege, PPG signals can be fed into the CNN in a usual 1D time-series signal or in an encoded 2D image. We will leverage three state-of-the-art 2D CNN architectures (VGG16, ResNet50, and MobileNetV2) that are pre-trained on huge dataset and transfer the extensive knowledge gained from other image classification tasks by repurposing the models and fine tuning them for our problem.

We train a separate model for each feature extracted from the PPG window individually. The input to each model is a five-minute PPG signal, and the output is a label obtained via the automatic PPG annotation method. An overview of the CNN-based approaches is shown in Figure 4.5. This shows that the five-minute PPG time-series and one of the HR-HRV features are fed to 1D CNN in the training phase. Moreover, the pre-trained 2D CNN models (VGG16, ResNet50, and MobileNetV2) are fed with the five-minute PPG images – encoded using Gramian Angular Field (GAF) – along with one of the HR-HRV features. In the following, we describe the CNN-based approaches with different architectures leveraged to perform PPG signal quality assessment.

4.5.1 1D CNN

We design a customized CNN method trained with the PPG time-series segments as having two convolution layers with one-dimensional filters trailed by a non-linear ReLU activation unit. Our convolutional layers are followed by a batch normalization layer [70], by which the

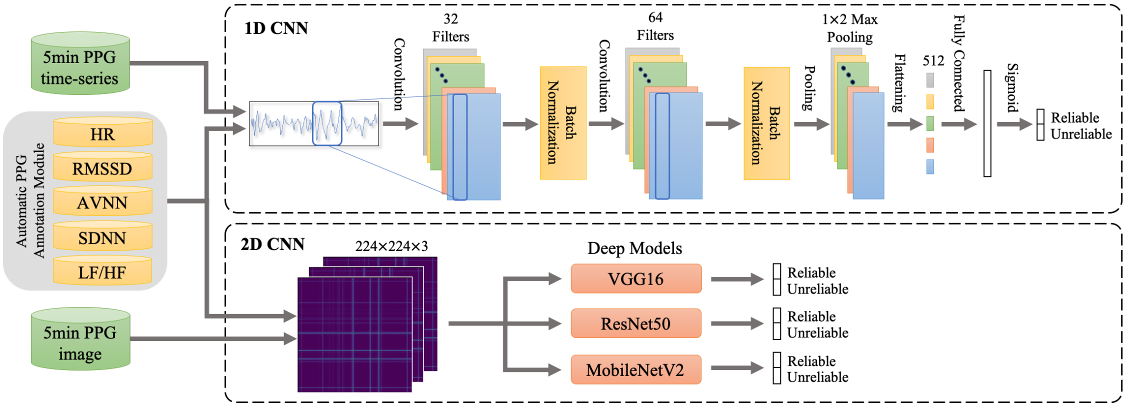


Figure 4.5: Overview of the CNN-based architectures for the PPG Quality Assessment

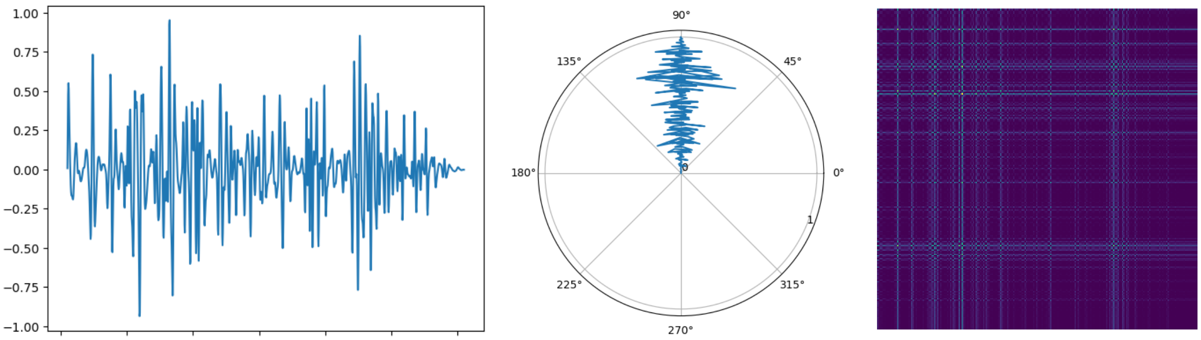
changes in the hidden unit values are reduced. Moreover, the batch normalization reduces overfitting since it has a slight regularization effect. The output of the convolution block is then fed into a maxpooling layer to reduce the dimensionality of the data. The learned features obtained through the convolutional block are flattened to one long vector and pass through a fully connected neural network before the output layer used to make a prediction. The fully connected layer ideally provides a buffer between the learned features and the output with the intent of interpreting the learned features before making a prediction. For our customized model, we will use a standard configuration of kernel size of 3 and 32 and 64 parallel feature maps for the first and second convolutional layers, respectively. The feature maps are the number of times the input is processed or interpreted, whereas the kernel size is the number of input time steps considered as the input sequence is read or processed onto the feature maps.

A grid search is also carried out to optimize the hyperparameters of the model. The efficient Adam version of stochastic gradient descent with a learning rate of 0.00001 is used to optimize the network, and the binary cross entropy loss function is used given that we are learning a binary-class classification problem. We analyze the performance of the customized 1D CNN model trained separately on every single feature of HR, AVNN, RMSSD, SDNN, and LF/HF to describe the reliability and unreliability of the signal with respect to that feature. For the training phase, five-minute PPG signals along with the label for each specific feature are fed to the 1D CNN.

4.5.2 2D CNN

We utilize three powerful pre-trained CNN networks, VGG, ResNet, and MobileNet and repurpose them and fine tune them with the PPG images. To create suitable inputs for the CNN, the PPG time-series need to be converted to PPG images, while preserving the temporal dependency of the time-series. Therefore, we encode the PPG signals as images using the Gramian Angular Field (GAF) method [192]. The GAF also contains temporal correlations –similar to an image– making the image proper for the CNN. The GAF is created based on the Gram Matrix defined by the dot product of every couple of vectors. The dot product shows the similarity of the set of vectors. Since the Gram Matrix produces a matrix with the size of time-series squared, we need to reduce the dimensionality of the input to decrease the amount of computation. In this regard, we use Piece-wise Aggregate Approximation (PAA) to reduce the dimensionality of the input time-series by dividing them into equal-sized, non-overlapping windows and extract the average in each segment [82]. To build GAFs, PPG as a time-series signal is scaled into [-1,1] with a Min-Max scaler. Next, PPG length is decreased using the PAA algorithm to a 224×224 image. Then, PPG is converted into the polar coordinates rather than keeping it in the typical Cartesian coordinates. The polar encoding is then followed by a Gram Matrix like operation on the resulting angles. Three stages of the PPG encoding is shown in Figure 4.6.

To leverage the significant performance of CNNs, we test our approach with three architecture: VGG, ResNet, and MobileNet. VGG stacks the convolutional layers with an increasing number of filters but with the same size of 3×3, since two 3×3 filters almost cover a 5×5 filter and are also more lightweight in multiplications [166]. Among VGG architectures, VGG16 and VGG19 are the most popular, since both followed the same strategy. VGG19



(a) Scaled PPG time-series (b) Polar Encoding (c) Gramian Angular Field

Figure 4.6: Encoding PPG time-series to PPG image with Gramian Angular Field (GAF)

is deeper than VGG16, but it is observed that the accuracy is not improved and saturated. The other architecture that we investigate is ResNet [66]. The success recipe of ResNet for training a deep network is the residual connections, where each layer is connected not only to the previous layer but also the layer behind the previous layer. With this intention, each layer has more information. The last architecture considered in this study is MobileNet [152]. MobileNet brought a novel idea of replacing a standard convolution with a depthwise convolution, followed by a pointwise convolution. This way of convolving performs a solitary convolution on every color channel as opposed to joining every one of the three and smoothing it. This way of convolving helps to build a smaller model and smaller complexity, which makes it suitable to run on mobile devices. In this paper, we implement the three 2D CNN models, VGG16, ResNet50, and MobileNetV2, and train them for each HR-HRV feature to describe the reliability and unreliability of the signal with respect to that feature. We pass the encoded PPG images as 2D inputs to these models along with the corresponding label obtained via the automatic PPG annotation method of each feature.

4.6 Experimental Results

In this section, we investigate the performance of the four approaches (i.e., one 1D CNN and three 2D CNN) described in Section 4.5. The approaches are utilized to classify the PPG signals according to the reliability of five PPG-based parameters: i.e., HR, AVNN, RMSSD, SDNN, and LF/HF ratio. In this regard, four models are trained for each health parameter, resulting in a total of 20 models.

Moreover, we evaluate the performance of our method in comparison to existing methods. In this regard, our best models for the health parameters are compared with six different PPG signal quality assessment methods. In the following, we first outline our setup and the data used for the training and testing the models. Then, we evaluate and compare the models in terms of their performance.

4.6.1 Setup

In our setup, we use Keras Sequential API from tensorflow to train and evaluate the CNN models [1]. We need to pass a couple of parameters to the Keras API including an optimizer, a loss function, and metrics that will be used in the training and validation phase of the model. Adam optimizer [85], binary cross-entropy loss as a loss function, and accuracy, f1-score, and area under curve (AUC) metrics are used to compile our CNN models. Our dataset has skew over one label, much more samples are from Reliable class for most of the classification models, which makes our classification problem to be an imbalance binary classification. Therefore, during the training and validation phase, we monitor the training and validation AUC and loss in each epoch [167]. The model with the highest AUC for the validation set during the optimization process is selected as the best model. That model will be the checkpoint for next epochs.

4.6.2 Training and Test Data Distribution

A total of 36 subjects are recruited for this study. We split the dataset into independent train set, validation set, and test set. 20% of the whole dataset (7 subject) is used for test set, and the rest is split into training and validation set, 80% (23 subject) and 20% (6 subject), respectively. For the sake of fair comparison, the models are trained and validated on the same train and validation dataset. To evaluate the performance of the DL models, we use an independent test set. The detail of the distribution of the train, validation, and test set is shown in Table 4.3.

Table 4.3: Distribution of train and test dataset on each annotation label

	Train (23) + Validation (6)		Test (7)	
Total Segments (# of Subjects)	58588		14628	
	Reliable	Unreliable	Reliable	Unreliable
HR	46175	12413	11544	3104
AVNN	50925	7663	12732	1916
RMSSD	16327	42261	4082	10566
SDNN	25822	32766	6456	8192
LF/HF ratio	42388	16200	10597	4050

4.6.3 Proposed Method Evaluation

As described in Section 4.5, four CNN models are trained for each PPG parameter: i.e., HR, AVNN, RMSSD, SDNN, and LF/HF. For each parameter, the model with the maximum AUC score is chosen. In the following, we evaluate the models on the validation set and test set. Finally, the best CNN models on the test set are selected to be used for the comparison with the stat-of-the-art methods.

Table 4.4: Validation Set performance results of different CNN models for various HR-HRV features

Label	HR			AVNN			RMSSD			SDNN			LF/HF		
Metric	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC
Model															
1D_CNN	**95.63**	**96.11**	**96.21**	**96.71**	**97.68**	**97.71**	**91.42**	**91.48**	**91.69**	**96.01**	**96.97**	**97.09**	**97.71**	**97.71**	**97.71**
MobileNet	94.58	93.93	94.63	95.68	95.95	96.93	89.68	89.44	89.46	92.66	93.91	93.66	91.92	91.95	92.93
ResNet50	92.49	92.44	93.52	90.43	90.22	91.32	86.47	85.81	87.17	92.52	92.12	93.66	90.22	90.22	91.31
VGG16	92.38	94.42	94.87	93.36	94.29	94.85	85.72	85.09	85.79	93.42	93.82	93.99	94.36	94.29	93.99

HR Models

The overall performance of our proposed CNN models created using the automatic annotated label for the HR feature is shown in Table 4.4. Classification accuracy, f1-score, and ROC area of all DL models for the test set is shown in Figure 4.7a. It can be seen that for the HR feature the **1D CNN** model works the best among all the DL models with an accuracy of 95.63%, F1-score of 96.10%, and AUC of 96.21%. As a result, the assessment of the reliability of PPG signal w.r.t. HR can be performed with high confidence using a one dimensional CNN classifier which is less complex compared to the two dimensional models.

AVNN Models

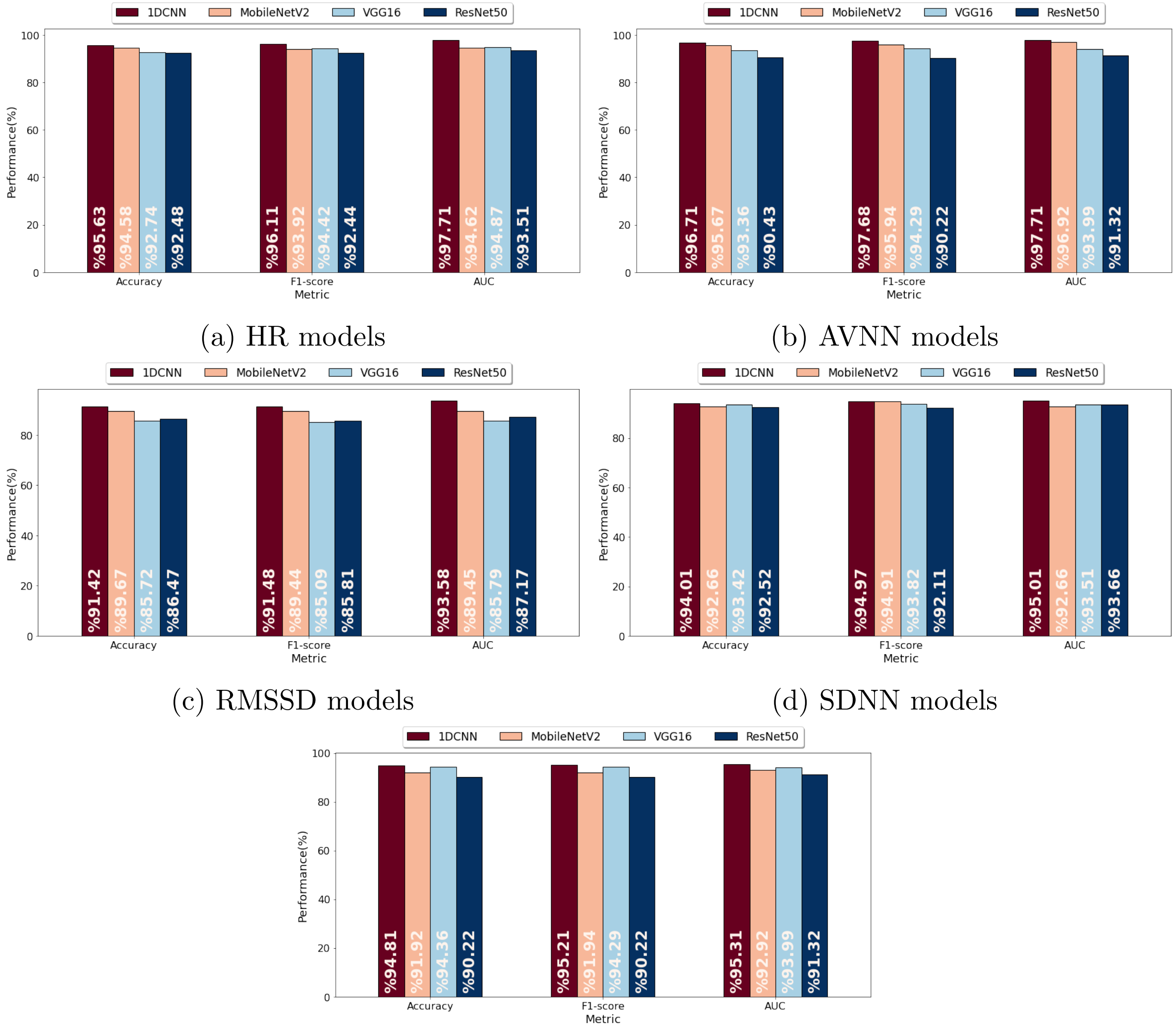
Table 4.4 summarizes the overall performance of our proposed CNN models created using the automatic annotated label for the AVNN feature, and Figure 4.7b shows the classification accuracy, f1-score and ROC area of the DL models for the test set. It can be seen that for the AVNN feature, the DL models show promising performances in all the metrics. However, the **1D CNN** model outperforms the other DL models with an accuracy of 96.71%, f1-score of 97.71%, and AUC of 97.71%. Following these results, quality assessment of PPG signals w.r.t. the AVNN can be done using a one dimensional CNN classifier with a marginal error.

RMSSD Models

The overall performance of the proposed CNN-based models for RMSSD is shown in Table 4.4. The classification accuracy, f1-score, and ROC area of the DL models for the test set are shown in Figure 4.7c. The DL models, built for the RMSSD feature, show a decent performance classifying the PPG signals as Reliable or Unreliable. For this feature, the 2D models show a mediocre performance comparing to the 1D CNN model. The performance of the RMSSD feature is worse than HR and AVNN due to the distribution of Reliable and Unreliable labels, which can be found in Table 4.3. Unlike HR and AVNN models with a very high Reliable to Unreliable class ratio, RMSSD models have the lowest Reliable to Unreliable ratio. This makes it difficult for the optimization process to find the best model. The **1D CNN** model outperforms the other DL models with an accuracy of 91.42%, f1-score of 91.48%, and AUC of 91.68%. Consequently, PPG quality classification based on the RMSSD feature can still be performed using a one-dimensional CNN classifier.

SDNN Models

Table 4.4 summarizes the overall performance of the proposed models for the SDNN feature. Moreover, Figure 4.7d shows the classification accuracy, f1-score, and ROC area of the DL models for the test set. It can be seen that all the 2D DL models trained for SDNN have a promising performance. The **1D CNN** model is the best classifier among all DL models with an accuracy of 96.01%, f1-score of 96.97%, and AUC of 97.08%. Therefore, the reliability of PPG signal w.r.t SDNN can be assessed with high confidence using a one-dimensional CNN classifier, a smaller and simpler model than the 2D models.



(a) HR models

(b) AVNN models

(c) RMSSD models

(d) SDNN models

(e) LF/HF ratio models

Figure 4.7: Bar charts showing a comparison of Test Set performance metrics for different DL models

LF/HF Models

The performance of the proposed models for LF/HF is shown in Table 4.4 (validation set) and in Figure 4.7e (test set). As indicated, the 2D DL models trained for LF/HF parameter have a promising performance; however, **1D CNN** model outperforms the other DL models with an accuracy of 97.71%, f1-score of 97.71% and AUC of 97.71%. As a result, the assessment of the reliability of PPG signal w.r.t. LF/HF can be performed with high confidence using a one dimensional CNN classifier which is less complex compared to the two dimensional models.

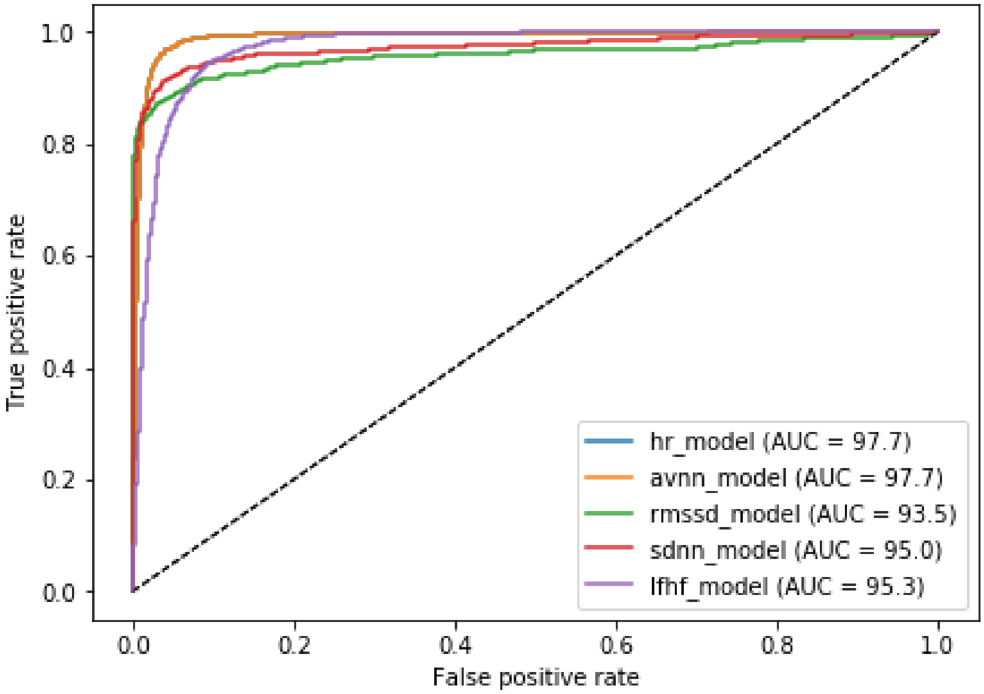


Figure 4.8: ROC curve of the Reliable class for 1D CNN models of all parameters

Figure 4.8 represents the Receiver Operating Characteristic (ROC) curves of the best model for each HR-HRV metric along with the AUC of the corresponding feature. It can be seen that different features have different characteristics in terms of PPG signal reliability. The **1D CNN** performs exceptionally well in classifying all HR-HRV features models into reliable and unreliable classes. In addition, **MobileNetV2** did also a great job for all HR-HRV models to classify them into reliable and unreliable classes. Our results show promising performance for the proposed CNN-based approaches, through which a binary decision is delivered to indicate PPG signal quality among five different features, HR, AVNN, RMSSD, SDNN, and LF/HF ratio in a real-time manner.

4.6.4 Comparison with State-of-the-art Methods

The results in Section 4.6.3 show that the **1D CNN** model outperforms the other DL models in terms of performance w.r.t to the HR and HRV features. Therefore, the proposed 1D CNN model is selected for comparison with the state-of-the-art models. As outlined in Section 4.2, there is a broad variety of PPG signal quality assessment methods in the literature. We compared our proposed method with six different methods. First, a rule-based method [187] is selected for comparison to distinguish low-quality signals, leveraging hierarchical decision rules combined with simple features, such as absolute amplitude and threshold crossing rate, and autocorrelation function features. Second, we compare the proposed method with Support Vector Machine, K nearest neighbors, and decision trees algorithms as supervised machine learning algorithms [128]. Furthermore, we compare our proposed method with an unsupervised method using an elliptical envelope [104]. Finally, our proposed method is compared with Xception [126] as a deep learning approach.

Table 4.5 shows the performance of the proposed method and the aforementioned state-of-the-art algorithms. The state-of-the-art quality assessment algorithms are only defined for the PPG signal itself and the HR feature. However, our proposed method can assess the quality of the PPG signals based on HR and HRV features using separate models. To make a fair comparison, the labels of the state-of-the-art algorithms are created using our automatic annotated labeling method described in Section 4.4.4. The proposed method (with the 1D CNN architecture) results in a more accurate signal quality assessment compared to the state-of-the-art. As indicated in the table, the rule-based and the unsupervised algorithms had the lowest overall performance. On the other hand, supervised traditional machine learning methods, such as KNN, showed promising results for AVNN and RMSSD. The accuracy of KNN for the RMSSD feature was slightly better than the proposed method. However, for the other features/metrics, the proposed method performs considerably better. The Xception algorithm, as a deep learning approach, had better performance, in general,

Table 4.5: Comparison between the proposed method and state-of-the-art on Test Set

Label	HR			AVNN			RMSSD			SDNN			LF/HF		
Metric	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC
model															
Rule Based[187]	61.19	77.28	67.80	63.08	72.44	68.35	52.13	59.95	61.81	63.08	75.44	71.35	67.65	76.15	65.70
KNN[127]	87.56	92.20	79.66	92.47	95.71	80.69	**93.70**	88.62	92.01	85.78	83.60	85.40	79.56	86.23	72.38
SVM[127]	78.70	88.08	50.00	86.84	92.95	50.00	89.54	79.52	84.46	78.39	75.84	78.24	72.42	84.00	50.00
DT[127]	78.70	88.08	50.00	86.84	92.95	50.00	89.54	79.52	84.46	78.39	75.84	78.24	72.42	84.00	50.00
Eliptical Envelope[104]	71.54	81.94	57.39	78.89	87.85	53.64	67.35	41.28	59.33	61.76	56.54	61.19	62.54	74.14	53.10
Xception[126]	89.30	90.28	85.80	89.54	90.44	85.35	81.03	70.95	77.81	88.54	90.44	85.35	85.09	88.15	84.70
Proposed Method	**95.64**	**96.12**	**97.71**	**96.71**	**97.68**	**97.71**	91.43	**91.48**	**93.59**	**94.02**	**94.98**	**95.02**	**94.82**	**95.22**	**95.31**

than the traditional supervised machine learning methods. It was able to pursue a stable performance of around 90%. However, the proposed method outperformed the Xception algorithm. On the other hand, the rule-based method [187] is the only algorithm that requires no training phase.

Chapter 5

Affective Signal Processing Pipeline

The first step in building a multimodal pain assessment system before feeding the collected biosignals into a classifier for pattern recognition, is to process the raw signals collected during trials due to the following reasons: 1) The contaminations in the raw signal could be prominent or even cover the useful information in the signal and therefore should be removed. 2) The parameters in a multi-parameter model could be sampled at different time intervals and unifying the time interval via feature extraction can simplify the modeling process. 3) The constructed model is more interpretable when the inputs are meaningful features instead of waveform segments. Data processing pipeline consisted of the following steps:

- We filtered the signal to remove powerline interference, baseline wander, and motion artifact noise.
- We performed feature extraction on the filtered signals to obtain amplitude, time, and frequency domain features. The time domain features were extracted using 5.5 second windows for the sake of comparison with the state of the art [194]. In addition to handcrafted features, we also used automatic features which were outputted from a

deep neural network.

- Once the features were extracted, we tagged them with their corresponding labels based on the nearest timestamp within 5.5 seconds of the label.

Each of these processing steps were applied individually to each of the modalities. Processed data from each of the modalities were fed to the feature extraction module to extract handcrafted and automatic types of features from each individual modality. The feature vectors then were combined using either early fusion or late fusion (explained in detail in the next chapter). This chapter presents a complete detail on the efficient affective signal processing pipeline block from Figure 1.1. The types of handcrafted features extracted from each modality and the deep learning pipeline for extracting automatic features are described in detail.

5.1 Hand-crafted Feature Extraction

The input data are processed in order to find and use patterns for predicting a latent pain state that the observed person is in. For this purpose, features, which are a more discriminative and usually lower dimensional representation, are extracted from the raw input data. Features may be categorized as (1) hand-crafted features and (2) automatic features. Handcrafted features are developed for the specific task taking advantage of human expertise; they are usually easy to interpret and lower dimensional. Examples are facial distances in image-based expression analysis or heart rate variability features extracted from ECG. With automatic features, the extraction of features are integrated into the modeling process. Most artificial neural network and deep learning approaches fall into this category. The automatic features are usually high-dimensional and not easy to interpret, but facilitate highest recognition performance if trained with enough suitable data (which however may be not available).

Generally, higher-dimensional feature vectors may contain more information, but also require more training data in order to identify the patterns that are relevant for the prediction task. Feature extraction is one of the key steps in the data analysis process which largely impacts the success of any subsequent statistics, and information is not expected to be lost in this stage [61] (" It is always better to err on the side of being too inclusive rather than risking to discard useful information. "). On the other hand, the complexity of the pattern increases with a high dimension and noisy features, which leads to feature selection in the next stage. In this work, the feature extraction centers on the experiential ones introduced in Chapter 2, for example, heartbeat interval and the sEMG features that can quantify muscle activity.

5.1.1 Electrocardiogram

An electrocardiogram (ECG) shows the strength and timing of electrical activity in the heart by measuring from specific sites on the body's skin surface. Figure 5.1 presents the shape of standard ECG wave in once cardiac cycle. Interpretation of ECG in pain studies starts with the ability to detect the QRS complex as one of the morphological parts in the center of an ECG cycle and one of the most visually distinct part, with the focus on the RR intervals (distance between adjacent R-waves). The correct detection of the R-peak in the QRS complex is the base of the following interpretive analysis such as heart rate extraction and heart rate variability (HRV) analysis in disease diagnosis, well-being tracking as well as in research studies where autonomic nervous system activities are observed [50, 161, 197].

Data Synchronization

The ECG signals from each patient were sampled at a rate of 500 Hz. Data from two channels (left arm, right arm) were obtained. The patient's pain levels were simultaneously reported and saved as labels. For the purpose of synchronicity, the corresponding Unix

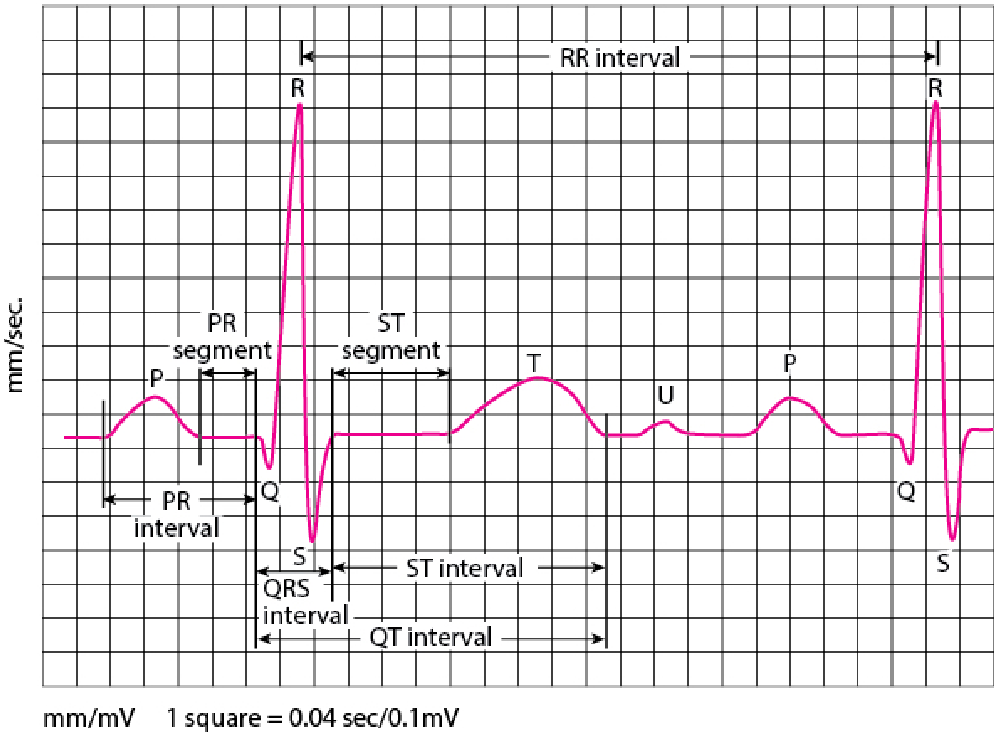


Figure 5.1: A sample of an ECG wave [31]

timestamps were also obtained while extracting both ECG and label data. The ECG signals were trimmed from start to end to match the corresponding label timestamps. Since the sampling frequency was 500 Hz, each timestamp had 500 ECG samples associated with it.

Peak Detection

The ECG channels were filtered using a Butterworth bandpass filter with the frequency ranges of [0.1,250] Hz. Once clean ECG signals were obtained, the second step in the pipeline was to extract peaks. To find the peaks, the signals were first sampled down to 250 Hz. A bidirectional long short-term memory network was used to obtain the probabilities and locations of peaks [90]. A window size of 1000 samples and stride of 100 samples was used to generate these predictions. Mean values were obtained from predictions that came from overlapping windows. The predictions that were below a particular threshold (0.05) were discarded and filtered out. Only those peaks that were in local maximum were selected.

Once the peaks were obtained, the signal was resampled back to 500 Hz, and the peak probabilities and locations were obtained. This method, however, might still be susceptible to false positives that are likely generated due to the presence of noise or irregular heartbeats. Therefore, another preprocessing step that removes peaks that occur too close to each other was employed. A rolling window was used to remove peaks that occurred in a time period of 450 milliseconds or less between neighboring peaks. The final selected peaks were then appended with their corresponding Unix timestamps. This process was repeated for every patient.

Noise Removal

The third and final preprocessing step is to remove noise from the NN interval data. NN intervals are the time intervals between two successive peaks. They are obtained by subtracting two successive peak indices. All data points that are within 2 standard deviations of the mean were selected. The rest of the data points were considered outliers and were removed. Even after removing these outliers, however, there might still be anomalous (not a number [NaN]) values in noisy sections of the data. If the proportion of NaN values exceeded 50 percent, the noisy sections were discarded. Otherwise, only NaN values were discarded, and the remaining values were interpolated. The filtered NN intervals were then saved and used for feature extraction.

Heart rate variability feature extraction

HRV is used to describe the phenomenon of oscillations between consecutive instantaneous heartbeats. HRV features have been traditionally calculated over a short period of 5 minutes or over a long period in 24 hours. However, in some cases such as acute physiological changes, some HRV features in less than a minute are also taken on as ultra–short-term analysis.

Summaries of HRV metrics and norms methods can be found in a comprehensive guideline in [29] and in a more recent review [161]. In the literature, HRV analysis has been examined as one of the main ways to measure pain in different types; in Sesay et al[160], for instance, with the majority of 120 patients, it was observed that regarding acute pain after minor surgery, NRS was correlated with low-frequency (LF) band and the ratio of LF to high-frequency (HF) band but not with HF. To monitor the nociception level of patients with multiple physiological parameters, HF in a 1-minute window was calculated in Ben-Israel et al [20]. Jiang et al [71] experimented with the correlation of HRV features in the ultrashort term with acute pain. They suggested that multiple HRV features can indicate the change from no pain to pain. Werner et al [194] compared no pain among pain levels 1-4 using a random forest (RF) classifier. They reported the detection of pain using a set of features from ECG signals.

We extracted ECG handcrafted features (i.e., heart rate variability (HRV)) from a 5.5 seconds window to make our results comparable to state-of-the-art. The HRV handcrafted features were extracted with pyHRV, an open-source Python toolbox [55] using the R-peaks extracted from the ECG signal via a bidirectional long short-term memory network [90]. There were 19 time-domain and 13 frequency-domain extracted features.

Time Domain Features The time domain (TD) features extracted from NN intervals, or the time interval between successive R-peaks, comprised of the slope of NN intervals, 5 NN interval statistical features, 9 NN interval difference features, and 4 heart rate features. These features were computed using 5.5-second sliding windows. The definitions of these time domain features are mentioned in Table 5.1

The breakdown of the 19 aforementioned features is explained as follows: slope of NN intervals—a polynomial fit of degree 1; 5 NN interval features—total count, mean, minimum, maximum, SD; 9 NN interval difference features—mean difference, minimum difference, maximum difference, SD of successive interval differences, root mean square of successive interval

Table 5.1: Time domain heart rate variability features and their definitions.

Feature	Description
HR (ms)	Beats per minute
AVNN (ms)	Mean of NN intervals
SDNN (ms)	Standard deviation of NN intervals
RMSSD (ms)	Root mean square of successive NN interval differences
NNXX (ms)	Number of NN interval differences greater than the specified threshold
pNNXX (%)	Percentage of successive NN intervals that differ by more than XX ms

Table 5.2: Frequency domain heart rate variability features and their definitions.

Feature	Description
VLF power (s^2)	Absolute power in very low-frequency band ($\leq$0.04)
LF power (s^2)	Absolute power in low-frequency band (0.04-0.15)
HF power (s^2)	Absolute power in high-frequency band (0.15-0.4)
LF peak (Hz)	Peak frequency in low-frequency band (0.04-0.15)
HF peak (Hz)	Peak frequency in high-frequency band (0.15-0.4)
Total power (s^2)	Total power over all frequency bands
LF/HF (%)	Ratio of LF-to-HF power

differences, number of interval differences greater than (a) 20 milliseconds and (b) 50 milliseconds, and percentage of successive interval differences that differ by more than (a) 20 milliseconds and (b) 50 milliseconds; and 4 heart rate features—mean, minimum, maximum, and SD. The FD features extracted via estimating the power spectral density (PSD) comprised of total power, 4 High Frequency (HF) band fast fourier transform (FFT) features, 3 very low frequency (VLF) band FFT features and FFT ratio of HF and low frequency (LF) bands. This resulted in 32 features in total.

Frequency Domain Features There were 13 frequency domain features extracted by estimating of power spectral density using the Welch method. These features were computed using 250-second rolling windows with a minimum threshold of 50 values per window. The definitions of these frequency domain features are mentioned in Table 5.2

The breakdown of the 13 frequency domain features is explained as follows: total power—total spectral power over all frequency bands; 4 HF band fast Fourier transform (FFT) features—peak, absolute, relative, normalized; 4 LF band FFT features—peak, absolute, relative, normalized; 3 very low-frequency (VLF) band FFT features—peak, absolute, relative; and FFT ratio of HF and LF bands.

Feature Selection

To ensure generalization and avoid overfitting, it is important to perform feature selection. This, in turn, reduces computational complexity and the time for training and validating models. Feature selection models can be placed under three broad categories: filter-based methods, wrapper-based methods, and embedded methods. Filter-based methods statistically determine the relationship between input variables (features) and the target variable (label). They provide a metric for evaluating and filtering out features that will be used by the model. They are also computationally cheaper than the other two methods and have a reduced risk of overfitting [177]. Among the filter-based methods, Gini impurity/information gain is a widely used method to select the most informative features for a classification problem. Usually, a decision tree–based model like an RF classifier is used to output a feature importance vector. Every node of a decision tree represents a condition on how to split values present in a single feature. In this process, similar data on the condition variable end up on the same side of the split. The splitting condition is based on impurity of the features chosen in every node. During the training process, how much each feature contributed to the decrease in impurity is calculated, and features are then ranked based on this measure.

5.1.2 Electromyogram

An electromyogram (EMG) measures muscle response or electrical activity in response to a nerve's stimulation of the muscle. EMG signal has capability of providing useful information by extracting features within various domains. These features fall into the following categories: amplitude, frequency, stationarity, entropy, linearity, variability, similarity and heart rate variability [200]. Among all the proposed sEMG features, no single one is unanimously recommended as the best representative of sEMG signal across applications and scenarios.

The preprocessing phase of EMG channels comprised of a 20Hz Highpass filter and two notch filters at 50Hz and 100Hz all using a Butterworth filter. We extracted EMG handcrafted features from a 5.5 second window. The EMG features were extracted using a library written by myself based on the information from Zhang Ph.D. dissertation [200]. Ten features were extracted based on the amplitude of the signal, four based on frequency, three based on entropy, and four based on variability. All 21 aforementioned features were calculated for 5 different EMG channels resulting in 105 EMG features. Table 5.3 lists the names and desciprtions of the involved sEMG features in our study.

5.1.3 Electrodermal Activity

EDA measures skin conductance derived from microscopic changes in the level of perspiration on the surface of the skin [25]. Sweat secretion is involved in a number of regulatory processes [47], and an abundance of research suggests that skin conductance responses (SCR) are associated with emotional arousal [47, 7, 41, 17]. Detection of SCRs are a result of the fluctuation of two underlying components of EDA activity: the slow and steady baseline tonic component, and the faster or reactive phasic component. Sudden shifts in phasic activity above tonic activity are known as EDA Peaks. When these peaks occur in response to stimuli, they are known as Event-Related Skin Conductance Responses (ER-SCR). When these peaks do not appear to be related to the presentation of a stimulus, they are referred to as Nonspecific Skin Conductance Responses (NS-SCR) [25, 41]. Figure 5.2 presents a sample EDA signal including tonic and phasic components along with the signal peaks. Assessing the number of EDA peaks provides quantitative input into the level of emotional arousal or intensity during an experimental session.

EDA Feature Extraction

Figure 5.3 shows our pipeline architecture for preparing the data and extracting the set of features for classification. There are 3 different sections in this pipeline: (1) Data Preparation, (2) pyEDA [10], (3) Post Feature Extraction.

Data Preparation The primary purpose of the Data Preparation in our pipeline is to synchronize the data with the labels. To prepare the data for feature extraction, we extracted the original signals' slices that match with their corresponding labels. With this aim, the slices of GSR data and their labels are collected in this part to be fed to the pyEDA for pre-processing and feature extraction.

Table 5.3: sEMG features and their definitions in Amplitude, Frequency, Entropy, and Variability domains

Feature	**Definition**	**Mathematical Description**
peak	peak	maximum value of the signal
p2pmv	peak to peak mean value	difference between the average of the local maxima and the mean of the local minima of the signal
rms	root mean square	$\sqrt{\frac{1}{n}\Sigma_{i=1}^{n}x_i^2}$
mlocmaxv	mean of local maxima values	$\frac{1}{n}\Sigma_{i=1}^{n}loc\_max_i$
mlocminv	mean of local minima values	$\frac{1}{n}\Sigma_{i=1}^{n}loc\_min_i$
mav	mean of absolute values	$\Sigma_{i=1}^{n}\lvert x_i\rvert$
mavfd	mean of the absolute values of the first differences	$\frac{1}{n-1}\Sigma_{i=1}^{n-1}\lvert x_{i+1} - x_i\rvert$
mavfdn	mean of the absolute values of the first differences of the normalized signal	$\frac{1}{n-1}\Sigma_{i=1}^{n-1}\lvert \tilde{x}_{i+1} - \tilde{x}_i\rvert$
mavsd	mean of the absolute values of the second differences	$\frac{1}{n-2}\Sigma_{i=1}^{n-2}\lvert x_{i+2} - x_i\rvert$
mavsdn	mean of the absolute values of the second differences of the normalized signal	$\frac{1}{n-12}\Sigma_{i=1}^{n-2}\lvert \tilde{x}_{i+2} - \tilde{x}_i\rvert$
zc	zero crossings	number of times that signal changes sign
fmode	mode frequency	mode of f (frequency spectrum of signal)
fmean	mean frequency	$\Sigma_{i=1}^{M}f_iP_i/\Sigma_{i=1}^{M}P_i$
fmed	median frequency	$\Sigma_{i=1}^{MDF}P_i = \Sigma_{i=MDF}^{M}P_i = \frac{1}{2}\Sigma_{i=1}^{M}P_i$
aprox	approximate entropy	reference to [200]
sample	sample entropy	reference to [200]
spectral	spectral entropy	reference to [200]
var	variance	$\frac{1}{n}\Sigma_{i=1}^{n}(x_i - \tilde{x})^2$
std	standard deviation	$\sqrt{var}$
range	range	Max - min
intrange	interquartile range	$Q_3 - Q_1$

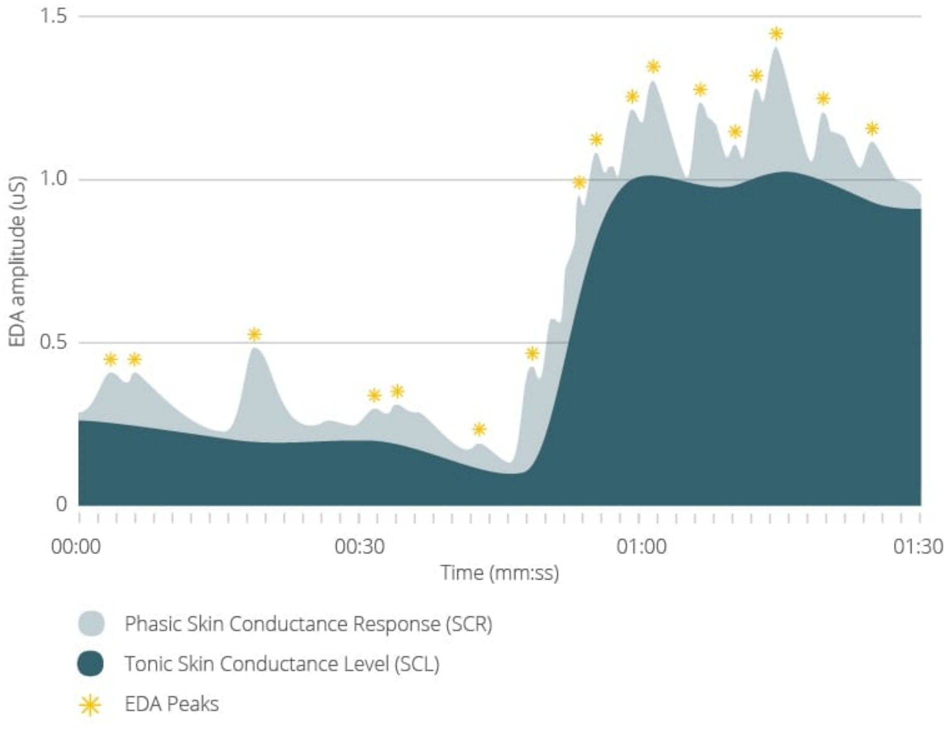


Figure 5.2: A sample of an EDA wave with tonic and phasic components

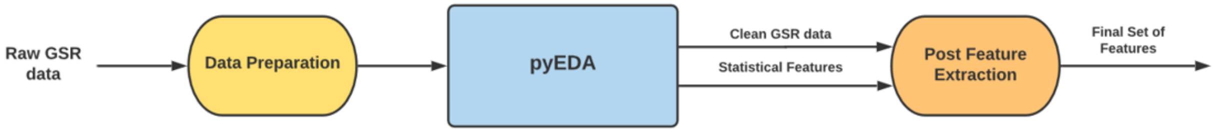


Figure 5.3: EDA feature extraction pipeline architecture

pyEDA The architecture of the pyEDA is shown in Figure 5.4. According to this figure, Preprocessing and Feature Extraction are the 2 main stages in this pipeline.

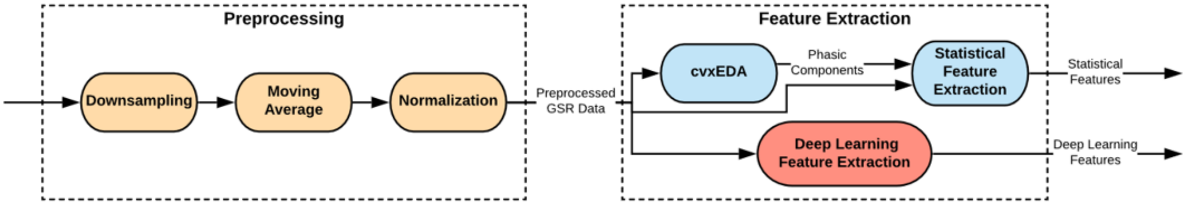


Figure 5.4: Pipeline architecture of the pyEDA

In the preprocessing stage of the pyEDA pipeline, at first, the data are down-sampled; then, a moving average is used to smooth the data and reduce the artifacts such as body gestures and movements. In the end, the data are normalized to become suitable for classification models.

If the GSR data are collected at 128 Hz, it can safely be down-sampled to a 20 Hz sampling rate. This down-sampling has been done to conserve memory and processing time of the data. In this work, we did not down-sample the data since the original data are already sampled at 4 Hz, which is good in terms of time and memory usage.

In this work, several steps were taken to remove motion artifacts from the GSR signal. First, we used a moving average across a 1-second window to remove the motion artifacts and smooth the data. Second, a low-pass Butterworth filter on the phasic data was applied to remove the line noise. Lastly, preprocessed GSR signals corresponding to each different pain level were visualized to ensure the validity of the signals.

The pyEDA uses 2 different algorithms for feature extraction (Statistical Feature Extraction and Deep Learning Feature Extraction). The parameters of the Deep Learning Feature Extraction part of the pipeline are set and tuned for stress detection; therefore, in this work, we only used the features extracted by the Statistical Feature Extraction algorithm.

The number of peaks, the mean, and the max peak amplitude are the 3 different statistical features that are extracted in the pyEDA. The GSR signals consist of 2 main components: skin conductance level, also known as the tonic level of GSR, and skin conductance response, also called the phasic component of GSR. The GSR peaks or bursts are considered the variations in the phasic component of the signal. Therefore, the most important part in extracting the peaks of the GSR signal is to extract its phasic component. Based on Figure 5.4, the pyEDA tool uses the cvxEDA algorithm [56] to extract the phasic component. Then, the phasic component and the preprocessed GSR data are fed to the Statistical Feature

Extraction module to extract the 3 mentioned features (number of peaks, mean GSR, and max peak amplitude).

Post Feature Extraction We also extracted the features that were used in the work by Werner et al [194] for the EDA signals. The preprocessed EDA signals and the set of features (number of peaks, mean EDA, and max peak amplitude) were fed into the Post Feature Extraction module to extract these features.

The maximum value of the peaks, range, standard deviation, interquartile range, root mean square, mean value of local maxima, mean value of local minima, mean of the absolute values of the first differences, and mean of the absolute values of the second differences are the extra features that were extracted in this part. Table 5.4 shows all the extracted features with their descriptions.

The mean of the absolute values of the first differences (mavfd) is calculated as:

$$mavf(x) = \frac{1}{N-1} \Sigma_{i=1}^{N-1} |x_{i+1} - x_i| \tag{5.1}$$

The mean of the absolute values of the second differences (mavsd) is calculated as:

$$mavs(x) = \frac{1}{N-2} \Sigma_{i=1}^{N-2} |x_{i+2} - x_i| \tag{5.2}$$

5.1.4 PPG-based Respiratory Rate

We pre-processed the PPG signal before extracting the respiratory rate from it. Two filters were used during the pre-processing. We first used a Butterworth bandpass filter to remove noises including motion artifacts. Then, a moving average filter was implemented to smooth the PPG signal. After that, we applied an Empirical Mode Decomposition (EMD) based

Table 5.4: Extracted electrodermal activity (EDA) features with their descriptions

Feature	**Description**
Number of peaks	The number of peaks
Mean	The mean value of the signal
Max	The maximum value of the peaks
Range	The difference between the maximum and the minimum value of the signal
STD	Standard deviation of the signal
IQR	The difference between upper and lower quartiles of the signal
RMS	Root mean square of the signal
Mean minima	The mean value of local minima of the signal
Mean maxima	The mean value of local maxima of the signal
mavfd	The mean of the absolute values of the first differences
mavsd	The mean of the absolute values of the second differences

method proposed by Madhav *et al.* [103] to derive respiration signals from filtered PPG signal. This method was proved to derive respiratory rate from a PPG signal with high accuracy (99.87%). Figure 5.5 shows a filtered PPG signal and its corresponding respiratory signal.

Respiratory Feature Extraction

As can be seen from Figure 5.5, the respiratory signal derived from a PPG signal only include inhale peaks, which is different from a regular respiratory signal. Therefore, extracting other types of respiratory features from this signal is not feasible. The respiratory features extracted in this study are briefly described in Table 5.5. A total of 10 features were extracted from the respiratory signal based on statistical measures of the filtered signal and the peak amplitudes.

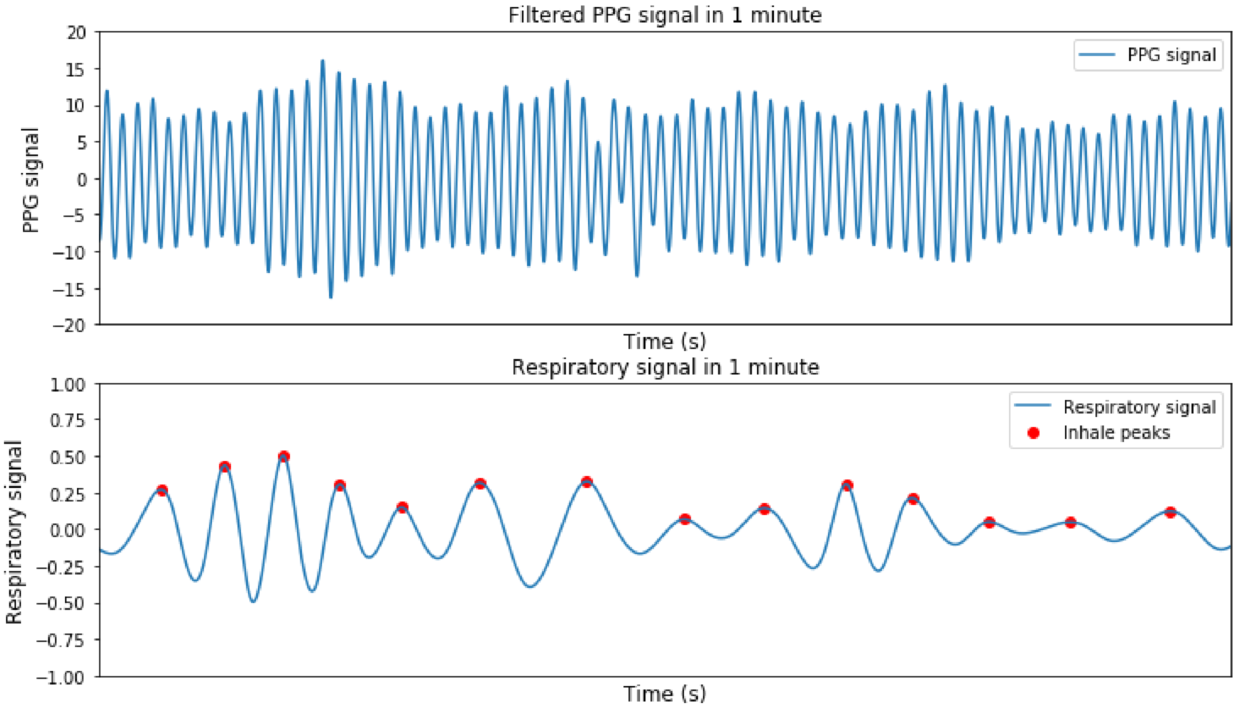


Figure 5.5: Filtered PPG signal and the corresponding respiratory signal in one minute

Feature Selection

To ensure generalization and avoid overfitting of the pain assessment model, we implemented a feature selection method. Feature selection reduces the training time and overfitting and improves the accuracy of the classification. In this study, we used a filter-based feature selection method as it is less computationally intensive and has a lower risk of overfitting compared with other methods [143]. This method statistically determines the relationship between input features and target labels. Gini impurity gain is used in our filter-based method to select the most informative features for the classification model. A decision-tree based random forest classifier is used to output the feature importance vector. Inside the decision tree model, every node is a condition on one of the features, and these nodes supposed to separate the data into two different sets. The data with the same labels will be separated into one group in an optimal scenario. The splitting condition depends on the impurity of the features chosen in every node. During the training process, the contribution

Table 5.5: Extracted respiratory rate features with their descriptions

Feature	Description
Peaks	The number of inhale peaks
Mean	The mean value of the signal
Max	The maximum value of the signal
Min	The minimum value of the signal
Range	The difference between the maximum and the minimum value of the signal
STD	Standard deviation of the signal
AVPI	The average value of the inhale peak intervals
SDPI	The standard deviation of the inhale peak intervals
RMS	The root mean square of successive differences between adjacent inhale peak intervals
COE	Standard deviation of inhale duration/average inhale duration

to the decrease in the impurity of each feature is calculated. Then, the importance of features is ranked according to this measurement.

5.2 Automatic Feature Extraction Pipeline

As the dimensionality of biomedical data increases, it becomes increasingly difficult to train a machine learning algorithm on the entire uncompressed dataset. This often leads to a large training time and is computationally more expensive overall. One possible solution is to perform feature engineering to get a compressed and interpretable representation of the signal. Another alternative approach, however, is to use the compressed or latent representation of that data obtained from deep learning networks trained for that specific task. Using automatic features helps in dimensionality reduction and can provide us with a sophisticated yet succinct representation of the data that handcrafted features alone cannot provide. This automatic feature extraction is typically carried out by an autoencoder network, which is an unsupervised neural network that learns how to efficiently compress and encode the data into a lower-dimensional space [154, 91]. Autoencoders are composed of two separate networks, an encoder and a decoder. The encoder network acts as a bottleneck layer and maps the

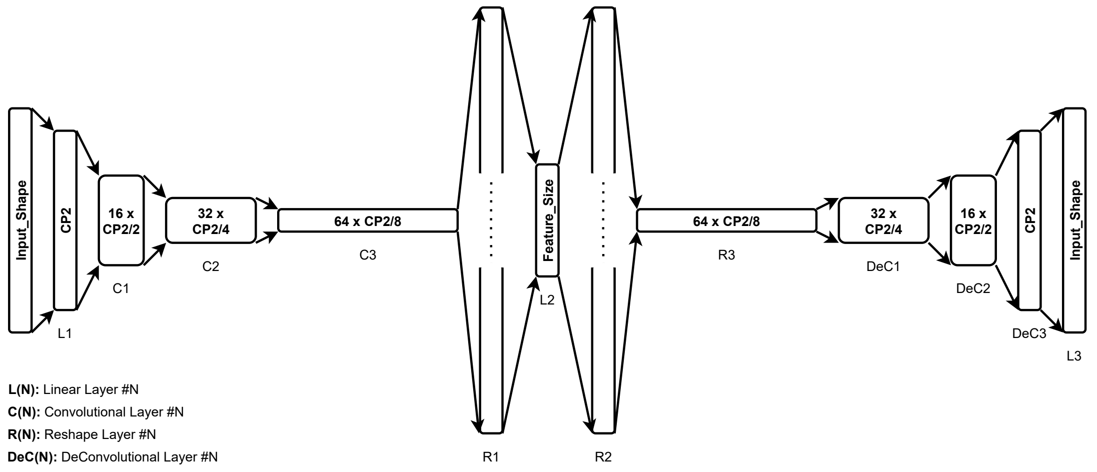


Figure 5.6: The architecture of the pyEDA convolutional autoencoder.

input into a lower-dimensional feature space. The decoder network tries to reconstruct this lower-dimensional feature vector into the original input size. The entire network is trained to minimize the reconstruction loss (i.e mean-squared error) by iteratively updating its weights and biases through back propagation.

A convolutional autoencoder from the pyEDA library was used to extract automatic features. Figure 5.6 shows the architecture of the autoencoder. First, a linear layer (L1) is used to down sample the input signal with Input_Shape length to a length that is the closest power of 2 (CP2). This was done to make the model scalable to an arbitrary input size. The encoder half of the network consists of three 1-D convolutional layers (C1, C2, and C3) and a linear layer (L2) which flatten and down samples the input vector to a lower-dimensional latent vector. The number of dimensions of this latent vector (Feature_Size) corresponds to the number of automatic features extracted and was set prior to training the network. A total of 32 features were extracted from ECG, EDA, and RR signals. Whereas, a total of 30 features were extracted from the EMG signal (6 features from each of the 5 channels). The dimension of the latent vector was chosen to be comparable to the number of handcrafted features extracted. The decoder half of the network consists of three 1-D de-convolutional layers (DeC1, DeC2, and DeC3) to reconstruct the input signal from the latent vector. A final linear layer (L3) is then used to flatten and reconstruct the signal to its original dimension. Both encoder and decoder networks have ReLU (Rectified Linear Unit) activation between layers. A window size of 10 seconds was applied to the filtered signals to provide the model with more temporal context. Furthermore, the input vector length using 10-second windows (as opposed to 5.5-second windows) seemed to generalize better among different sampling rates across modalities. After signals from each of the modalities were normalized, they were divided into 10-second chunks and trained on separate autoencoder models.

The batch size was set to 10, the number of training epochs was set to 100, and the ADAM optimizer [202] was used with a learning rate of 1e-3. A total of 126 feature vectors across all 4 modalities were extracted. A visualization of the automatic feature extraction pipeline is shown in Figure 5.7.

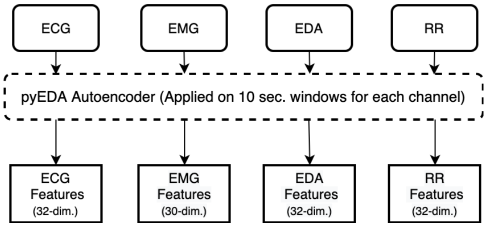


Figure 5.7: Automatic feature extraction pipeline.

Chapter 6

Pain Assessment from Physiological Features

The study design described fully in Chapter 3 will allow us to perform a validation on our proposed methods of pain assessment. In this chapter as presented in Figure 1.1, Pain Assessment with MMML techniques block, we discuss challenges and solutions in the NRS label distribution recorded during our clinical trials. Furthermore, we use the collected data to study each modality and perform various method validations, unimodal and multimodal, predictive models. We evaluated our models using leave-one-subject-out cross-validation.

6.1 Label Augmentation

There were a number of inherent challenges in the distribution of labels as NRS values recorded during the clinical trials of this study were collected from real postoperative patients. This problem bears less significance while studying healthy participants since the stimulated pain can be controlled during the experiments. As a consequence, occurrences of some pain levels far exceeded those of others. Figure 6.1 shows the distribution of 11 NRS pain labels reported by participants during the clinical trials. As can be seen from this figure, among all patients, there were only 4 reported occurrences of pain level 10, whereas there were more than 80 reported occurrences of pain level 4. This imbalanced distribution was inevitable and an inherent challenge for further classification due to the subjective nature and the different sources of pain among real post-operative patients during clinical trials.

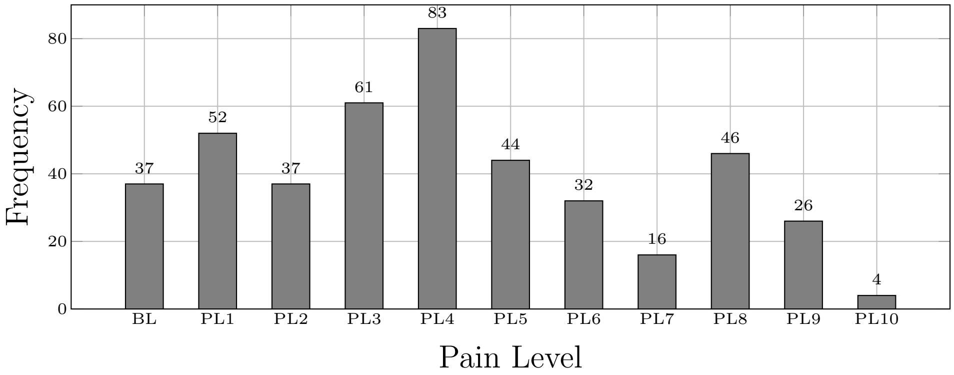


Figure 6.1: The distribution of 11 classes NRS labels

To compare our pain assessment algorithm's performance with state-of-the-art [194, 180], we downsampled our pain labels from 11 NRS classes (0-10) to 5 classes (0-4). Data points from NRS pain label 0 were considered as a baseline, and the remaining NRS pain labels were distributed among 4 classes. Thresholds for each downsampled class were carefully chosen to ensure a more evenly distributed set of labels and minimize the imbalanced class distribution. The pain levels ranged from a baseline level of pain (BL) or no pain to 4 increasing intensities of pain (PL 1-4). Figure 6.2 shows the resulting distribution after downsampling the NRS pain labels. The relatively large number of occurrences of NRS pain label 4 increased the number of downsampled PL2 labels over other downsampled pain labels.

Since we asked patients to report their pain levels only while they performed pain-inducing activities, the number of labels generated was sparse. The hand-crafted features were combined with the corresponding labels using timestamps that were within the nearest 5.5 seconds (labeling threshold) of the reported NRS value. The automatic features used a labeling

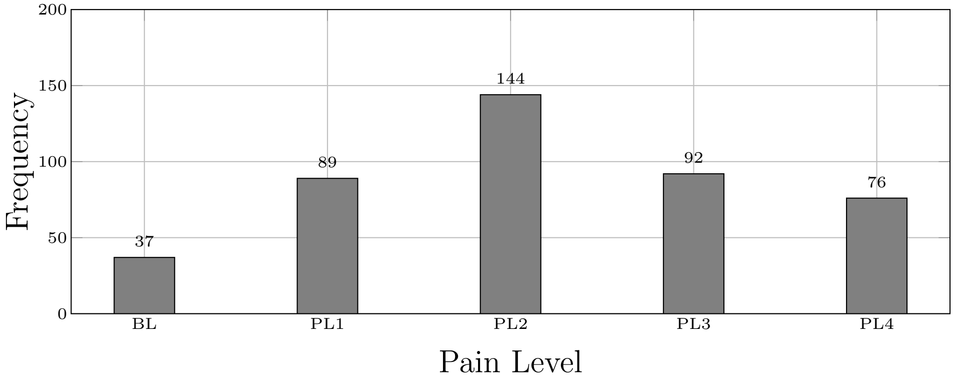


Figure 6.2: The distribution of 5 classes labels after using the Snorkel and downsampling

threshold of 10 seconds instead. As a consequence of having sparse labels, many of the feature windows were not assigned a corresponding label. To mitigate the problem of having an imbalanced and sparse label distribution, two techniques were exploited.

6.1.1 Minority Oversampling

The first technique, called Synthetic Minority Oversampling (Smote), is a type of data augmentation that over-samples the minority class [33]. Smote works by first choosing a minority class instance at random and finding its k nearest minority class neighbors. It then creates a synthetic example at a randomly selected point between two instances of the minority class in that feature space. The experiments involving Smote were implemented using the imbalanced-learn python library [94].

6.1.2 Weak Supervision

The second technique we utilized is weak supervision using the Snorkel framework [143]. Rather than employing an expert to manually label the unlabelled instances, Snorkel allows its users to write labeling functions that can make use of heuristics, patterns, external knowledge-bases, and third-party machine learning models. Weak supervision is typically

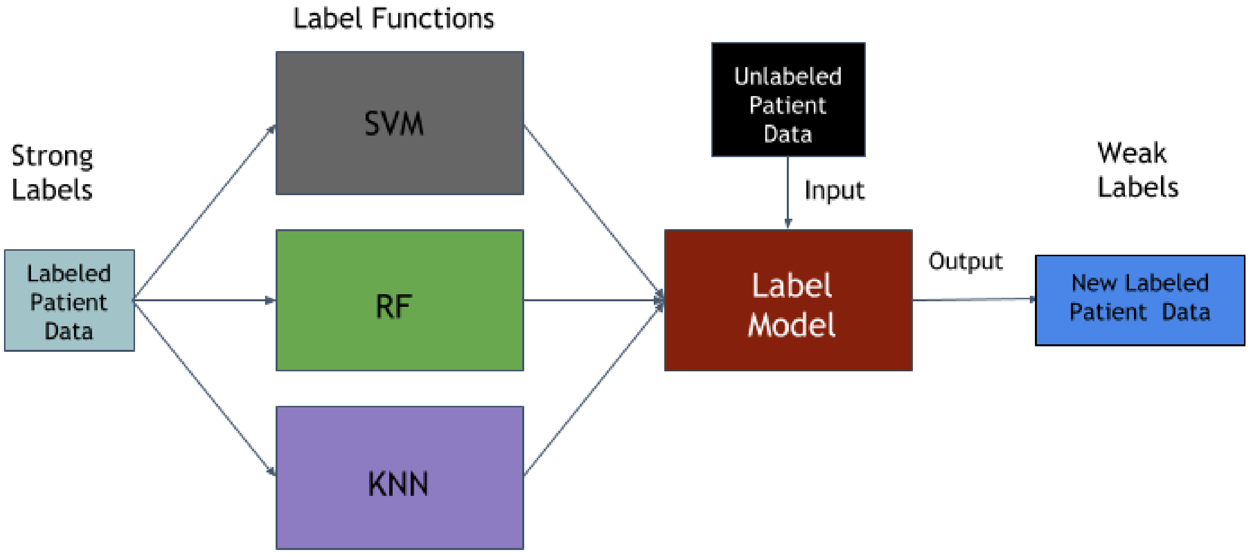


Figure 6.3: Snorkel labeling architecture

employed to label large volumes of unlabeled data when there are noisy, limited, or imprecise sources. This approach eliminates the burden of continuously obtaining NRS values from patients. The use of Snorkel in our labeling process allowed us to make use of more data for the purpose of training and testing our pain algorithm. This consequently led to better performance during validation. Figure 6.3 depicts the architecture that was used to label these unlabeled instances. For the purpose of our pain assessment algorithm, we decided to use third-party machine learning models to label the remaining unlabelled instances.

All the data points that were within the labeling threshold were considered as “strong labels”, or ground-truth values collected from patients during trials. The remaining unlabelled data points were kept aside for Snorkel to provide a weakly supervised label. The strong labels were fed into Snorkel's labeling function consisting of three off-the-shelf machine learning models: (i) a Support-Vector Machine (SVM) with a radial basis function kernel, (ii) a Random Forest (RF) classifier, and (iii) a K-Nearest Neighbor (KNN) classifier with uniform weights. Once each model was trained on the strong labels, it was used to make predictions on the remaining unlabeled data. The predictions from these three models were collected and converted into a single confidence-weighted label per data point using Snorkel's ”LabelModel" function. This function outputs the most confident prediction as the label for each data point. To perform a fair assessment of the reliability and accuracy of our algorithm, we used Smote and Snorkel only while training our machine learning models. The performance of these models was measured solely on ground-truth (strong) labels collected during trials. This way, there is no implicit bias introduced from mislabeling or up-sampling certain data points to skew model predictions.

6.2 Unimodal Machine Learning Models

6.2.1 ECG

Since the NRS labels recorded during clinical trials were collected from real postoperative patients, there are some inherent challenges due to the distribution of data. For example, there are 83 occurrences of NRS pain label 4, but there are only 4 occurrences of NRS pain label 10 among all patients. Due to the subjective nature and the different sources of pain among our recruited patients, the imbalanced distribution of pain levels among all patients is inevitable.

To compare our pain assessment algorithm's performance with Werner et al. [194], we downsampled our pain labels from 11 NRS classes (0-10) to 5 classes (0-4). Data points from NRS pain label 0 were considered as a baseline, and the remaining NRS pain labels were distributed among 4 classes. Thresholds for each downsampled class were carefully chosen in order to minimize an imbalanced class distribution. Table 4 shows the resulting distribution after downsampling the NRS pain labels. The relatively large number of occurrences of NRS pain label 4 increased the number of downsampled PL2 labels over other downsampled pain labels.

Table 6.1: Distribution of downsampled labels in the UCI_iHurtDB after down-sampling.

Pain level	Frequency, n
BL	37
PL1	89
PL2	144
PL3	92
PL4	76

Classification

To compare the performance of our pain algorithm with state-of-the-art [194], we performed binary classification on our test data using the hand-crafted features. We split the binary classification problem into 4 different categories: baseline (BL) versus pain level 1 (PL1), BL versus PL2, BL versus PL3, and BL versus PL4. Since one of the patients had data from only one downsampled label class, they were discarded from the classification process. Consequently, we were left with data from 19 patients.

We evaluated the performance of our pain algorithm using leave-one-out cross-validation (LOOCV) with the focus on optimizing the area under the curve (AUC) score. During each iteration of LOOCV, the data of 18 of the 19 patients, including those data points that were labeled by Snorkel, were used for training. For testing, only the strongly labeled data points from the one patient left out were used. This process is repeated for all 19 patients to estimate the algorithm's performance on unseen data. Due to the presence of an imbalanced distribution of pain levels within patients, data points from some pain levels were nonexistent from their data. As a result, it was not possible to compute either precision or recall for most patients.

The following five classification methods were deployed in our experiments to identify the best performing model for our pain assessment algorithm: AdaBoost classifier, XGBoost classifier, RF classifier, SVM classifier, and KNN classifier.

We also conducted separate experiments with feature selection using the 32 ECG hand-crafted features mentioned in the Section 5.1.1. To get the best set of features for classification, we run LOOCV using an RF classifier. We compute the Gini importance of each of the features at every fold and select those features that were at least one standard deviation above the mean importance score. As a result, it was possible to have different sets of features in every fold. After computing the best set of features at every fold, we consider those features that were used in most of the folds for classification. The following 8 features were used in the final feature set: (1) total spectral power, (2) absolute LF power, (3) absolute HF power, (4) mean HR, (5) relative HF power, (6) normalized HF power, (7) relative VLF power, and (8) normalized LF power.

6.2.2 EDA

Feature Selection

One of the key components in machine learning is to select the set of features that has the highest importance in classification. Performing feature selection on the data reduces overfitting, reduces training time, and improves accuracy. By removing the set of features that are not informative for our classification and only add complexity to our model, there is less opportunity to make decisions based on noise, making the model less over-fitted. Fewer data means less training time. In the end, by having more informative data and fewer misleading data, the accuracy of the model increases.

Random forests [30] are among the most popular machine learning methods. They provide 2 methods for feature selection: mean decrease impurity and mean decrease accuracy. In this work, we used a mean decrease impurity method for feature selection.

Mean decrease impurity is also sometimes called Gini importance. Random forest is an ensemble learning algorithm consisting of several decision trees. The decision tree is a tree-like model of decisions in which every node is a condition on one of the features. These nodes separate the data into 2 different sets so that in the optimal scenario, the data with the same labels end up in the same set. Impurity is the measure based on which the optimal condition is chosen on every node. Mean decrease impurity for each feature is defined as the total decrease in node impurity averaged over all ensemble trees. The features are ranked according to this measure.

Labeling the features

In the work by Werner et al. [194], there were 5 different pain levels, including the baseline level. To properly compare our pain assessment algorithm with their work, we down-sampled our 11 classes to 5 classes. The key factor in this down-sampling is to ensure that the distribution of the labels is as balanced as possible. As a result, we considered pain levels 1-3 as new pain level 1 (PL1), pain level 4 as new pain level 2 (PL2), pain levels 5-7 as new pain level 3 (PL3), and pain levels 8-10 as new pain level 4 (PL4). Based on Table 2, there are only 37 data points for the baseline. To increase the number of samples for the baseline to make our labels more balanced, we up-sampled PL0 based on the reported PL0 data by the patients. We ensured these new baseline data were close enough to the reported pain level 0 labels (less than 10 seconds difference) and had no overlap with other labels. These assumptions were made to make sure (1) we were not reproducing any data and (2) the patients had the same pain level 0 for these new timestamps. By doing this procedure for all the participants, our number of samples for pain level 0 increased from 37 to 86.

Table 6.2 shows the distribution of the down-sampled labels and the new baseline. The distribution of the new labels is appropriately balanced. Still, for PL1, the number of samples is slightly higher than the rest of the classes. This is because we down-sampled our pain levels to 4 different classes to make our settings comparable with the work by Werner

Table 6.2: Distribution of labels in the UCI_iHurtDB after down-sampling.

Pain level	**Frequency, n**
BL	86
PL1	150
PL2	83
PL3	92
PL4	76

et al [194].

Classification

We used machine learning–based algorithms to evaluate the performance of our pain assessment algorithm. Two different classification methods were used here: (1) k-nearest-neighbor with k between 1 and 20 and (2) random forest with a depth between 1 and 10. The k-nearest-neighbor method uses k number of nearest data points and predicts the result based on a majority vote [4]. The random forest classifier is an ensemble learning algorithm that fits several decision tree classifiers on various subsamples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting [26]. We used the Scikit-learn library to create our classification models [124]. Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression, and clustering algorithms, including k-nearest-neighbor and random forest.

To accurately evaluate the performance of our classification models, we used a cross-validation method [87]. Cross-validation is one of the most popular algorithms used to truly estimate a machine learning model's accuracy on unseen data. It achieves this by training a model using different subsets of data and obtaining the average accuracy on the rest of the data as a test. In this work, we used leave-one-out cross-validation to evaluate our result. We considered all the data acquired from one of the patients as a test and created our pain model using the rest of the patients. We repeated this procedure for each patient as a test. Each time, we created our pain model from scratch without considering the current test patient data or any information from the previous pain models. The final accuracy of the model was obtained by averaging the accuracy of all constructed pain models.

6.2.3 PPG-based Respiratory

Feature Selection

To ensure generalization and avoid overfitting of the pain assessment model, we implemented a feature selection method. Feature selection reduces the training time and overfitting and improves the accuracy of the classification. In this study, we used a filter-based feature selection method as it is less computationally intensive and has a lower risk of overfitting compared with other methods [143]. This method statistically determines the relationship between input features and target labels. Gini impurity gain is used in our filter-based method to select the most informative features for the classification model. A decision-tree based random forest classifier is used to output the feature importance vector. Inside the decision tree model, every node is a condition on one of the features, and these nodes supposed to separate the data into two different sets. The data with the same labels will be separated into one group in an optimal scenario. The splitting condition depends on the impurity of the features chosen in every node. During the training process, the contribution to the decrease in the impurity of each feature is calculated. Then, the importance of features is ranked according to this measurement.

Features Labeling Method

Figure 6.1 shows the distribution of 11 NRS pain labels reported by participants during the clinical trials. As can be seen from this figure, unbalanced label distribution is an inherent challenge for further classification since these NRS labels were recorded from real post-operative patients during clinical trials. For example, there are 97 pain labels ”four”, but there are only 4 pain labels ”ten” among all patients. Such unbalanced distribution of pain levels is unavoidable because of the subjective and realistic nature of this study and the presence of different intensities and/or pain sources among the patients.

Respiratory signal related features are usually calculated per one minute. However, considering that the time interval of pain labels reported by patients might be less than one minute, we chose 20 seconds as the feature extraction window to avoid data overlap during the labeling process. This indicates that all the pain labels are matched to their nearest 20 seconds feature window. After that, we used the Snorkel weak supervision method [143] to label other feature windows for which the NRS pain labels does not exist. Snorkel is an end-to-end method that leverages a weak supervision method to label a training dataset when limited ground truth data is available. This method is in particular beneficial for our case to address the unbalanced nature of our labels collected in a real setting. In this study, we considered all the NRS pain labels collected directly from the patients as ”strong” labels to train the labeling function. Each patient’s strongly labeled data was only used to mark their unlabeled windows. These remaining data points labeled by Snorkel are called ”weak” data. We only use these weakly labeled data in our training process (not in the validation process) to ensure a fair evaluation of the pain assessment accuracy. In other words, our algorithm’s final performance is assessed using only real data collected from post-operative patients. After the labeling process, the number of each pain label gained varying degrees of growth resulting in 58 pain levels ’0’ to use as the baseline.

To compare the performance of our pain assessment algorithm with the state-of-the-art [180], we downsampled the pain labels from 11 classes (0-10) to 4 classes (0 to 3). All the labels of pain level 0 were taken as the baseline while the remaining 10 classes were grouped into 3 classes. We considered pain level 1-3 as a new pain level 1 (PL1), pain level 4-5 as a new pain level 2, and pain level 6-10 as a new pain level 3 (PL3). These downsampling thresholds were chosen carefully to minimize the unbalanced label distribution problem. The new labels distribution is shown in Figure 6.4.

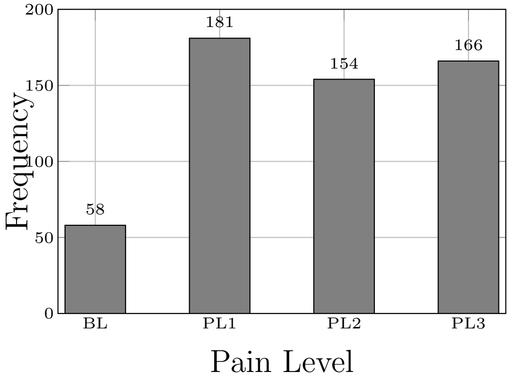


Figure 6.4: The distribution of 4 classes labels after using the Snorkel and downsampling

Classification

We used a machine learning based approach to build predictive models for pain assessment. Five different classification methods were implemented, including ADABoost, XGBoost, random forest, support vector machine (SVM), and k-nearest neighbor (KNN) classifiers.

To evaluate the performance of our classification models in terms of generalizability, the Leave-one-subject-out cross-validation method was used. For each iteration of the cross-validation, we considered all the data points with only strong labels as the test set and trained our pain model using the data points including strong labels as well as weakly supervised labels using Snorkel for the rest of the patients.

6.3 Multimodal Machine Learning Models

To compare the performance of our multimodal machine learning models with the prior work, we performed binary classification using a leave-one-subject-out cross-validation approach [87]. In this method, a model's performance is validated over multiple folds in such a way that data from each patient is either in the training set or in the testing set. The purpose of using this method is to provide generalizability to unseen patients and to avoid overfitting by averaging the results over multiple folds. The eventual goal of this study is to build personalized models that make predictions on a single patient, but learn from data collected from a larger population of similar patients.

The following machine learning models were used to evaluate the performance of our pain assessment algorithm: (1) K-nearest neighbor with K ranging from 1 to 50, (2) Random Forest classifier with a depth ranging from 10 to 100, (3) AdaBoost (Adaptive Boosting) with the number of base estimators ranging from 20 to 2000, (4) and a Support Vector Machine (SVM) with a radial basis function kernel and a degree of 3. The optimal hyperparameter settings for these models were obtained using randomized grid search with a 3-fold cross validation. The best parameters were selected for each model and they were then evaluated using leave-one-subject-out cross-validation. Four separate models were trained for each of the four pain intensities (e.g., BL, no pain versus PL1, the lowest pain level, or BL vs PL4, the highest pain level).

6.3.1 Fusing Modalities

Two fusion approaches were used while combining features across different modalities. The first one being early or feature-level fusion which concatenates feature vectors across different modalities based on their timestamps. The resulting data that is now higher in dimension than any one single modality is then fed into our classifier to make predictions. While concatenating features across different modalities, a threshold of 5.5 seconds was used to combine all hand crafted features and a threshold of 10 seconds was used to combine the automatic features. There were a total of 159 and 126 different features amongst the handcrafted and automatic features, respectively. The second approach was late or decision level fusion where each modality is fed to a separate classifier and the final classification result is based on the fusion of outputs from the different modalities [60].

6.3.2 Feature Selection

Since there were a lot of features generated during the data processing phase, we had to select a subset of the most informative features to build our models with. Therefore, to reduce the complexity and training time of the resulting model, feature selection using Gini importance was performed. Gini importance is a lightweight method that is simple and fast to compute. Since we extracted a relatively large number of features in our method, it made sense to use a computationally low cost algorithm for feature selection. To obtain the best set of features for our classification models, we ran a leave-one-person-out cross-validation fold on the four different models for each pain intensity using an AdaBoost classifier. We computed the Gini importance of the features from the training data and selected the top n features (where n ranged from 10 to 50 in increments of 10). Since there were multiple folds, it was possible to have different sets of features for each of the folds. We considered the most commonly selected features across all folds as the final set of features to use for our model. In this way, each of the pain intensity models could have a different subset of features across each of the four modalities because these models operate independently of each other.

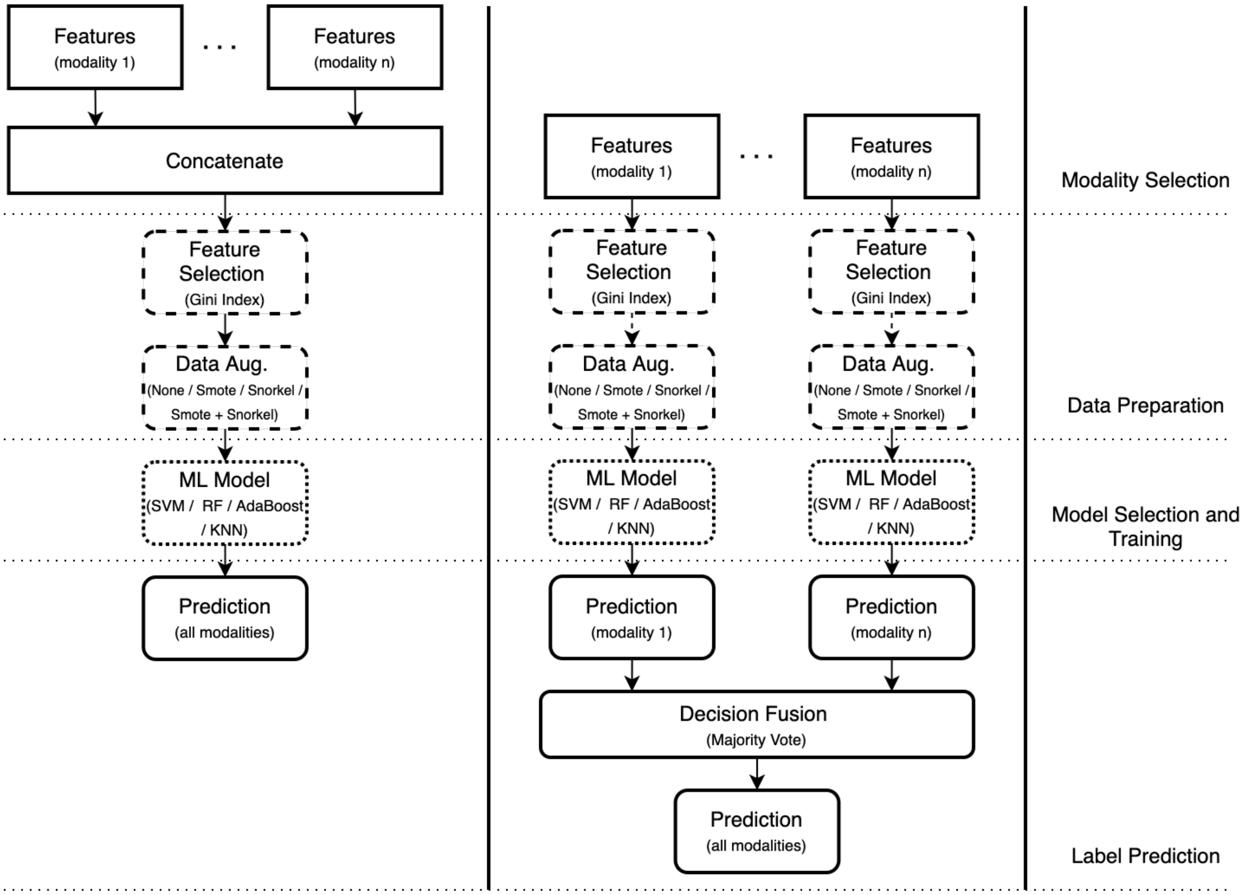


Figure 6.5: Our proposed general multimodal pipeline based on early fusion (left) and late fusion (right).

6.4 Experimental Results

6.4.1 Experimental Settings

The goal of our experiments was to compare the performance of using only a single modality to build our models over using a combination of multiple modalities. We trained several different models for each of the pain intensities that varied in the types of modalities, label augmentation techniques, machine learning models, and fusion techniques used. Figure 6.5 shows the general pipeline of the experiments we conducted. We first select the type of modalities to train on, which varied from only using each of the single modalities separately to using a combination of 3 or more modalities. Moreover, these modalities varied

Table 6.3: Validation accuracy of BioVid features.

Binary classification	AdaBoost	XGBoost	RF	SVM	KNN	Werner et al
BL vs PL1	52.63	41.35	42.97	69.16	39.06	48.7
BL vs PL2	75.68	69.57	70.84	84.14	70.92	51.6
BL vs PL3	66.33	65.73	65.94	75.73	64.20	56.5
BL vs PL4	41.53	44.55	44.24	62.72	44.68	62.0

on the types of features used, like handcrafted or automatic features. In the case of using multiple modalities, we had two choices of fusion; early (Figure 6.5 left) and late (Figure 6.5 right). These architectures varied in how the modalities were combined, either before training (early), or at decision level (late) after training using majority voting. The data preparation process involved feature selection and label augmentation. These models could either be trained with no label augmentation, with just Smote or Snorkel, or a combination of both of them. The last step of the pipeline before making predictions involved choosing the type of machine learning algorithms, like Support Vector Machine (SVM), Random Forest (RF), Adaptive Boosting (AdaBoost), or K-Nearest Neighbors (KNN). The best performing single and multimodal model configurations are mentioned in the section below.

6.4.2 Results

ECG

To make a fair comparison between our pain assessment algorithm and the work of Werner et al [194], we replicated their settings into our data set. The comparisons of the accuracy achieved by our algorithm on all five classifiers while using only 3 time domain features are shown in Figure 6.6. Similarly, Figure 6.7 shows the same comparison while performing feature selection. These figures show the mean accuracy across all subjects while performing 4 different binary classifications based on pain levels. The final scores are presented in Table 6.3 and Table 6.4 below.

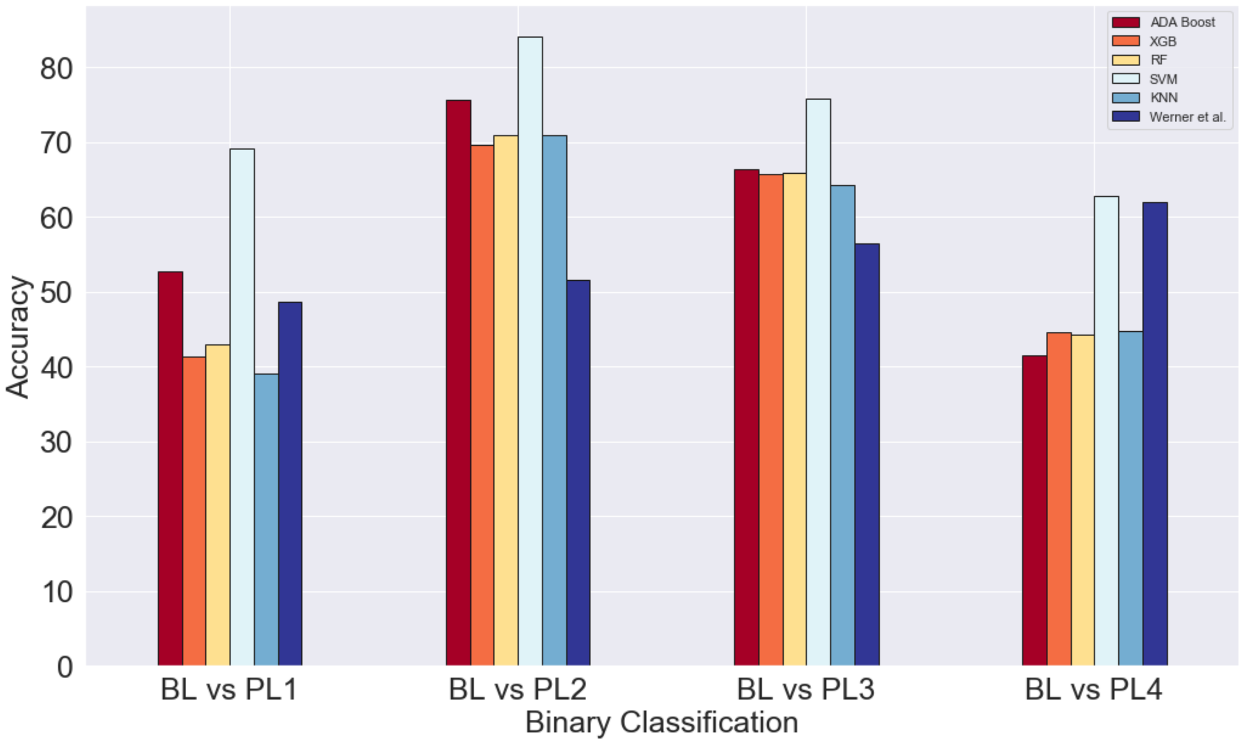


Figure 6.6: Validation accuracy of all classifiers on BioVid features

Table 6.4: Validation accuracy of top 8 features.

Binary classification	AdaBoost	XGBoost	RF	SVM	KNN	Werner et al
BL vs PL1	59.94	59.46	64.37	67.03	58.61	48.7
BL vs PL2	71.06	68.85	77.19	84.79	68.54	51.6
BL vs PL3	62.63	59.22	64.29	76.18	53.76	56.5
BL vs PL4	39.29	60.44	43.17	63.86	32.51	62.0

We were able to achieve the highest accuracy on the SVM classifier for both settings, with and without feature selection. Moreover, there is no noteworthy difference in the performance of the SVM classifier in both settings. However, while comparing the other classifiers, it is evident that there is a great improvement in performance while using feature selection in the BL versus PL1 category. The performances of the AdaBoost, XGBoost, RF, and KNN classifiers have a marked increase of about 12% on average when compared to their counterparts without feature selection. However, there is a slight decrease in the AdaBoost and RF classifiers and significant decrease in the KNN performance in the BL versus PL4

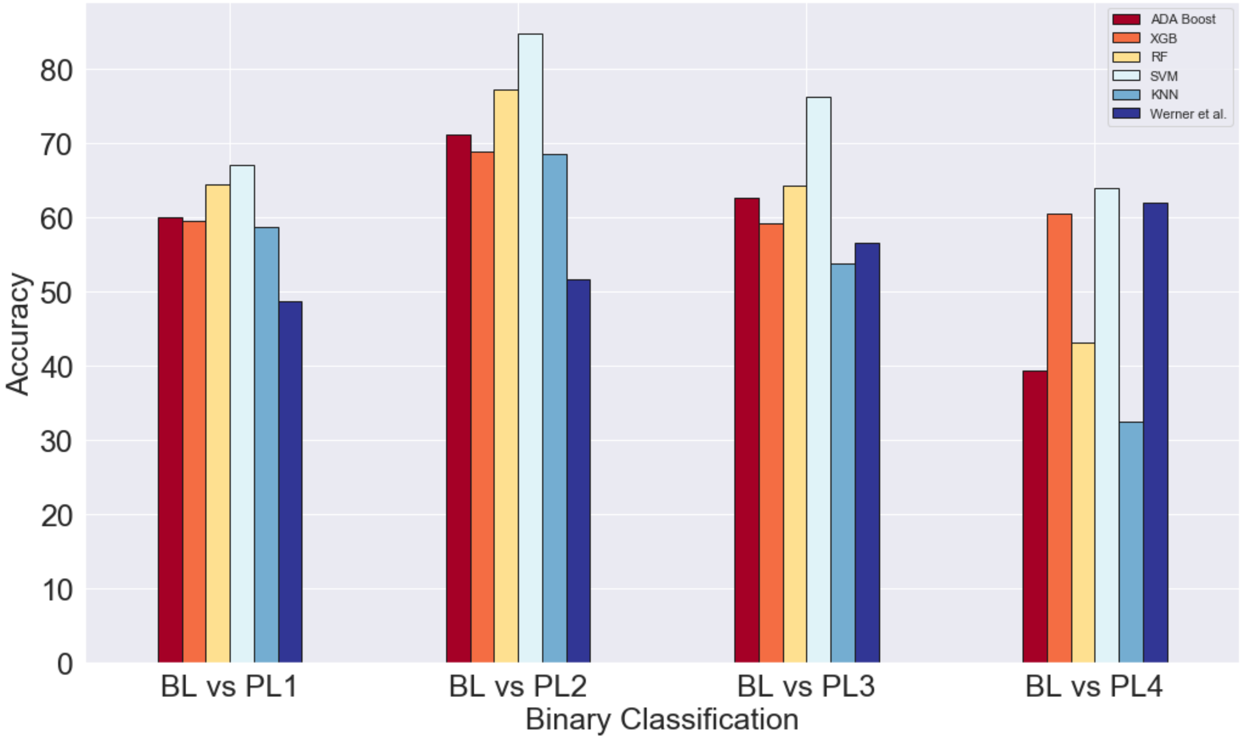


Figure 6.7: Validation accuracy of all classifiers on top 8 features

category. On the other hand, there is an improvement of about 16% in the XGBoost classifier while performing feature selection in the BL versus PL4 category. We speculate that the lower accuracy scores could be due to the relatively smaller number of training examples available from the downsampled PL4. On the flip side, due to the relative abundance of training examples from PL2, there is a spike in performance for all classifiers across both feature settings in the BL versus PL2 category.

While comparing our algorithm's performance to Werner et al [194], we can see that our SVM classifier fares significantly better than their model. The SVM classifier outperforms their model by an average of 20% across both feature settings for the first three pain categories (BL vs PL1, BL vs PL2, and BL vs PL3). Conversely, there is only a slight increase in performance across both feature settings in the BL versus PL4 category.

Table 6.5: Validation accuracy of BioVid features.

Binary classification	RF	KNN	Werner et al
BL vs PL1	84.0	74.4	55.4
BL vs PL2	66.3	67.5	60.2
BL vs PL3	57.2	65.0	65.9
BL vs PL4	55.2	53.0	73.8

EDA

To show that our pain assessment algorithm can achieve comparable results to the work by Werner et al. [194], we used identical settings as their work. Werner et al. [194] used 5 different pain levels, including the baseline. They also considered 5.5-second windows for the EDA data. Therefore, in the Data Preparation part of our pipeline, we considered 5.5-second windows for the slices of the EDA data for feature extraction (2.75 seconds before and after each timestamp). Furthermore, as discussed in the Methods section, we down-sampled the pain levels from 11 classes to 5 classes to make them similar with their labels.

At first, we used the set of features that was used in the work by Werner et al. [194] for classification without any feature selection. The maximum value of the peaks, range, standard deviation, interquartile range, root mean square, mean value of local maxima, mean value of local minima, mean of the absolute values of the first differences, and mean of the absolute values of the second differences are the features that were used here.

We used leave-one-person-out cross-validation using k-nearest-neighbor and random forest algorithms. We reported the accuracy based on 4 different pain models (BL vs PL1, BL vs PL2, BL vs PL3, and BL vs PL4). Table 6.5 shows the comparison of the validation accuracy achieved by our classifiers with that by the pain models of Werner et al [194].

According to these data, for the first 2 pain models (BL vs PL1 and BL vs PL2), we achieved a higher accuracy using both of our classifiers in comparison with Werner et al [13]. For the

Table 6.6: Selected set of features for each pain model.

Pain models	Set of features
BL vs PL1	Mean, max, RMSa, and mean maxima
BL vs PL2	Mean, max, RMS, and mean maxima
BL vs PL3	Max and RMS
BL vs PL4	Mean, max, IQR, RMS, and mean maxima

third pain model, our accuracy is also close to their models, with less than 1% difference using the k-nearest-neighbor classifier. For the fourth model, the accuracy of our models was noticeably lower than their models. As the next step, we added 2 more features (the number of peaks and the mean of the EDA data) to our set of features and then selected the most important ones using the mean decrease impurity method to improve the accuracy.

To obtain the best set of classification features, we ran leave-one-person-out cross-validation on different pain models using random forest classifiers. We computed the Gini importance of the features on the training data and selected the top k number of features for training the model and classification (2-11 were considered to be possible values for k). Since we had a different number of folds, we could have different sets of features for each fold. We considered the set of features that was used in most of the folds as the final set of features for the current pain model. Table 6.6 shows the selected features for each pain model. Descriptions for each of these features can be found in Table 5.4.

According to this table, the maximum value and root mean square of the signal are 2 features that were selected in all the pain models. The mean value of the local maxima and the mean value of the signal were also selected for all the pain models except the third one. The difference between upper and lower quartiles of the signal (IQR) is a feature that was selected as an important feature for classification only for the BL vs PL4 pain model.

After the set of features for each pain model were obtained, we ran one-person-leave-out cross-validation for k-nearest neighbor and random forest algorithms using these sets of new

Table 6.7: Validation accuracy of top 8 features.

Binary classification	RF	KNN	Werner et al
BL vs PL1	86.0	76.8	55.4
BL vs PL2	70.0	69.1	60.2
BL vs PL3	69.8	72.1	65.9
BL vs PL4	61.5	60.0	73.8

selected features to achieve the final results.

Table 6.7 shows the validation accuracy comparison of our models with those by Werner et al [13] using feature selection for 5.5-second windows. As shown in this table, by using feature selection, our classifiers' accuracy improved compared to those shown in Table 6.5. For all the pain models except the fourth one, we were able to achieve higher accuracy than the pain models by Werner et al [194]. For BL vs PL4, our accuracy was still about 10% less than their work. In the Discussion section, we explain the potential reasons for this difference in our model.

PPG-based Respiratory

8 out of 10 features, including MAX, MIN, Range, AVPI, Mean, STD and Peaks were selected by our feature selection model. The pain assessment results using 5 classifiers were shown in Figure 6.8 together with the results from Thiam et al. [180] for comparison. This figure's values are the average accuracy across all subjects resulted from performing three different binary classifications based on pain levels. The final scores are summarized in Table 6.8.

As can be seen from Figure 6.8, all of our five classifiers achieve a higher accuracy compared with results reported by Thiam et al., for the first two pain levels (BL vs PL 1 and BL vs PL 2). It should be noted that our models use only 8 features whereas there are 65 features used in the model proposed in [180]. As for pain level 3, only the random forest and SVM classifiers outperform their accuracy. Using the same random forest classifier on respiratory

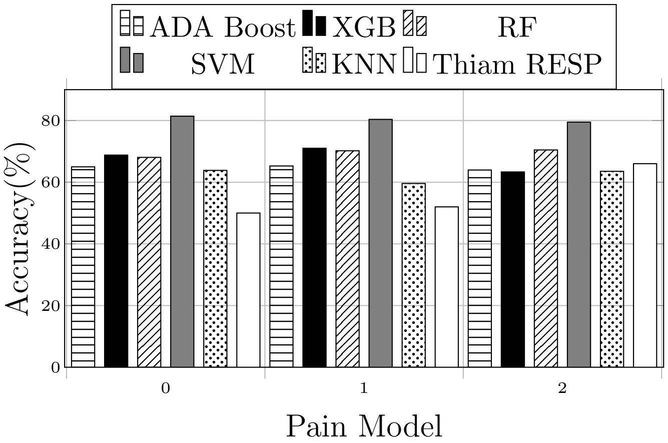


Figure 6.8: Validation accuracy of all classifiers on top 8 features

Table 6.8: Validation accuracy of our models in comparison with [180]

Pain levels	ADA Boost	XGB	RF	SVM	KNN	[180]
BL vs PL 1	64.98	68.79	68.04	**81.41**	63.82	50
BL vs PL 2	65.26	71.01	70.18	**80.36**	59.58	52
BL vs PL 3	63.95	63.33	70.45	**79.48**	63.52	66

features used by Thiam et al. (65 features), the accuracy of the three classifications is improved by 18.04%, 18.18%, and 4.45%.

Among the models, the SVM classifier achieved the highest performance. The accuracy of the three pain levels are 81.41%, 80.36%, and 79.48% separately. Compared to [180], the differences for three pain levels are 31.41%, 28.36%, and 13.48%, respectively. Our SVM classifier is significantly outperforms their model using only 8 respiratory features compared to the 65 features used in their models.

Single Modality vs. Multi-Modality

Tables 6.9 and 6.10 present the best performing single modal and multimodal models for each of the four pain intensities. The best multimodal models are mentioned along with their

Table 6.9: Single Modality: Best Scores

Pain Levels	ECG Scores	EMG Scores	EDA Scores	RR Scores
BL vs PL1	72.04	69.72	63.36	71.79
BL vs PL2	81.13	80.14	79.24	82.14
BL vs PL3	69.80	84.14	69.59	76.64
BL vs PL4	63.41	74.73	63.70	66.67
Mean	**71.59**	**77.18**	**68.97**	**74.31**

Table 6.10: Multiple Modality: Best Scores

Pain Levels	Werner *et al.*	Our Scores	Modalities	Classifier Config.
BL vs PL1	55.4	77.13	ECG, EMG, EDA, RR	Early Fusion + Auto. Feat. + Smot
BL vs PL2	60.2	85.64	ECG, EMG, EDA	Early Fusion + Auto. Feat. + Sm
BL vs PL3	67.7	86.90	ECG, EMG, EDA	Early Fusion + Auto. Feat. + Sno
BL vs PL4	77.8	74.73	EMG	Auto. Feat. + Snorkel +
Mean	**65.27**	**81.10**		

ML algorithm, label augmentation, and feature engineering methods employed to achieve their results. For comparison, the best multimodal results from Werner *et al.* [194] are also mentioned.

6.4.3 Discussion

From the single modality results (Table 6.9), it is evident that EMG models outperform all other modalities especially for the BL vs PL3 and BL vs PL4 models. Overall, models from all modalities have relatively lower scores in the BL vs PL1 and BL vs PL4 pain groups. On the contrary, the models for the middle two pain intensities performed better due to the relative abundance of such labels reported during trials. It should be noted that accuracy was used as a validation metric instead of F1 and AUC scores because a lot of the patients did not experience all of the pain levels (BL, ... PL4). As a consequence, we could not compute their true and false-positive rates. The comparatively lower performance of EDA models over other modalities suggests that variations in EDA signal response to different pain levels are more difficult to distinguish. Moreover, the EDA signals were collected using the Empatica E4 wrist worn device which makes them more prone to motion artifacts during trials. The multimodal results (Table 6.10) suggest that the best-performing models were built using automatic features. Most of these models used either one of the data augmentation techniques or a combination of both. The improved results obtained using such data augmentation like Smote and Snorkel are understandable due to the imbalanced distribution and sparse nature of labels collected from post-operative patients.

In terms of modalities, the best-performing models used EMG either alone or in combination with other signals. One justification for this could be due to the dynamic nature of EMG signals collected from facial muscles while experiencing pain. Since we were able to effectively isolate and capture periods of higher pain intensity with smaller window sizes, this could help the models better distinguish between baseline and other pain levels. This is especially evident in the BL vs PL4 models, where EMG alone provided the best results for both single modal and multimodal models.

The best-performing multimodal models use a combination of early fusion or feature level fusion along with a data augmentation technique. One intuition as to why early fusion might perform better overall is due to the detection of correlated features across modalities obtained after using feature selection [146]. Late fusion, on the contrary, builds independent models for each modality and fuses them based on their predictions using majority voting. Therefore, by treating each modality as independent, there is a potential loss of correlation in the combined feature space.

Overall, the multimodal models outperform all the single modal models in the first three pain intensities. It is clear that using multiple modalities enhances the models' ability to distinguish between different pain levels. The single modality results, however, can provide us with some key insights on which modality to prioritize in the absence of other modalities. While comparing our results to [194], we can see that our models outperform their models in all pain classes except the last one. This is understandable because our data was not collected from healthy subjects in controlled environments where the label distributions are more balanced.

Chapter 7

Intelligent and Adaptive Framework for Multi-Modal Machine Learning Affective Computing Services

Smart affective computing applications are fundamental to human experience, health and well-being and deliberately influence emotion or other affective phenomena by continuously monitoring physiological and contextual data. The connection between emotional states and physical health has become more known and has motivated the field of affective computing. Affective computing uses both hardware and software technology and multi-modal machine learning (MMML) kernels to analyze data form different sensor modalities, automate decision-making, and detect the affective state of a person. Input data perturbations (i.e., noisy inputs and motion artifacts) and varying system environment dynamics (i.e., network connectivity and signal strength) during sensory data acquisition affect the i) prediction accuracy and resilience of affective computing services, ii) energy efficiency in processing non-informative data, and iii) optimal sensing and sense-making choices. Monitoring raw data from sensor inputs to identify and drop non-insightful data and features from noisy modalities can improve prediction accuracy and energy efficiency. In this chapter, as presented in Figure 1.1, we present improving end-to-end system metrics of performance, energy consumption, and prediction accuracy of MMML-based affective computing services necessitates joint optimization of sensing and sense-making. We propose an adaptive *sensing* and an *intelligent sense-making* using reinforcement learning-based monitoring scheme for multi-modal affective computing applications that can learn to find the optimal feature and model set by i) monitoring input modalities, ii) analyzing input data to adaptively drop noisy data and feature, and iii) choosing the proper ML models to fit the selected feature vector per modality - to improve energy efficiency while providing sufficient prediction accuracy.

7.1 Introduction

Smart affective computing services deliver intelligent patient-centric digital healthcare and well-being services through continuous monitoring and analysis of multi-modal input data from physiological, and contextual sensors [114]. Smart affective computing services widely use multi-modal machine learning (MMML) algorithms to analyze input data from different sensor modalities, and generate accurate predictive results [113]. Design of end-to-end smart affective computing services can be split into two phases viz., *sensing* - acquiring input data from multi-modal sensors, and *sense-making* - analyzing input data through MMML algorithms for predictive results [112]. Efficiency of smart affective computing services is affected by – i) higher volumes of multi-modal input data from heterogeneous sensory devices, ii) compute intensity of MMML algorithms for data analysis, and iii) limited energy, and compute resources of wearable sensory devices and mobile computing nodes, particularly at the sensing layer [77]. Further, real-world affective computing applications are designed to monitor patients' symptoms in *everyday settings* through wearable sensors, and providing corresponding treatment plans. Typically, continuous longitudinal data acquired in *everyday* *settings* is prone to higher input data perturbations such as noisy components, unreliable signals, and motion artifacts, in comparison with the data acquired in clinical settings using reliable medical-grade sensors [198]. Processing input data with different perturbations affects the prediction accuracy, and model confidence of machine learning algorithms used in affective computing services [111]. Further, processing non-qualitative data incurs significant performance penalty, and energy drain, with resources spent on un-insightful computations [101].

Input data perturbations originating from the *sensing* phase influences the computational workload, prediction accuracy, model confidence, and energy consumption in the *sense-making* phase. Existing pre-processing, and data filtering techniques monitor the signal quality to selectively sense qualitative input data, and minimize noisy components [142]. Selective sensing aims at reducing the computational effort spent on noisy input data, and consequently reducing the energy consumption of both sensors, and processing elements [129]. State-of-the-art selective sensing techniques use data filtering and compression [142], context-aware sensing [129], and heterogeneity-aware sensing [117] to reduce the total input data volume. Despite the large body of recent work, existing optimizations for end-to-end smart affective computing services address multi-modal sensing, and sense-making dis-jointly [106, 19, 97].

Improving the system metrics of performance, energy consumption, and prediction accuracy necessitates joint optimization of multi-modal sensing and sense-making phases. The joint optimization approach allows *adaptive sensing* – to selectively extract reliable, and insightful input data, and features, and *intelligent sense-making* – to choose MMML models suitable for given input data, and feature set. Together, the adaptive sensing, and intelligent sense-making reduce computational workload, communication penalties, and energy consumption incurred in processing unreliable and/or low-quality data, and improve prediction accuracy, and resilience of affective computing services towards input data perturbations. However, co-optimization of multi-modal sensing, and sense-making phases requires *continuous monitoring* of sensor modalities to detect input data perturbations, *selective feature aggregation* to isolate reliable inputs, and *choice of suitable learning algorithms* for the given input modalities, and feature vectors.

In addition to input data perturbations, varying system environment dynamics such as network connectivity, and signal strength, user mobility, and available energy budget of sensory devices etc., further affect the optimal sensing, and sense-making configuration choices. Selecting the optimal sensing, and sense-making configuration settings, considering multi-modal input data perturbations, MMML algorithms, and varying system dynamics is an NP-hard problem [162]. Intelligence is essential to dynamically observe system dynamics to optimally configuring sense-making phase.

Understanding the underlying system dynamics (e.g., network condition), sensing variation (e.g., noisy sensing condition) and intricacies among computation, communication, accuracy, and latency is necessary to find optimal sense-making configuration in affective computing services. Reinforcement learning (RL) is an effective approach to develop such an understanding and interpret the varying dynamics of such systems [121]. RL allows a system to identify complex dynamics between influential system parameters and make a decision online to optimize objectives such as response time [175].

To this end, we propose two joint sensing and sense-making approaches, 1) an adaptive multi-modal sensing and sense-making, AMSER, and 2) an intelligent multi-modal sensing and sense-making approach, IMSER. They embed signal quality monitoring, adaptive sensing and an intelligent sense-making to orchestrate smart affective computing services for improving performance, energy consumption, and prediction accuracy. Figure 7.1 shows an overview of the proposed sensor-edge architectures for our intelligent and adaptive framework. The proposed system consists of multiple sensor devices at the sensor layer and computing resources at the edge layer. The sensor layer includes the multi-modal signal capability needed for smart affective computing services and the edge layer is a computing device providing data gateway, data processing, signal monitoring, adaptive sensing, intelligent sense-making controller, and inference at the edge. We designed a reinforcement learning orchestration scheme which performs adaptive feature and modality selection, machine learning model selection, and sensor configuration based on sensor modality monitoring at runtime to optimize energy consumption given accuracy requirements. We will discuss the two approaches in more details in the following sections.

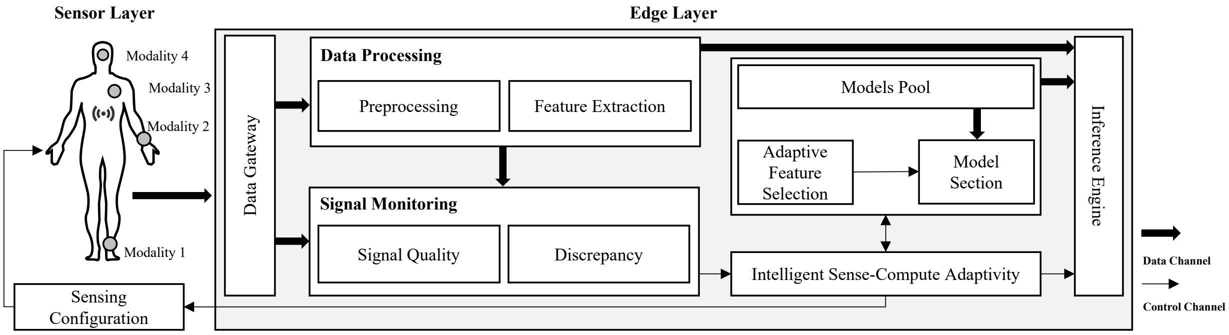


Figure 7.1: System Architecture Overview.

7.2 Related Works and Motivation

MMML-based affective computing services require to meet the following criteria: ***(a) Resiliency*** to recover from any perturbation and provide high quality of experience, ***(b) Energy-efficiency*** to enable long-term continuous monitoring on battery-powered wearable devices, and ***(c) Performance*** to deliver real-time and accurate services for rapid response to emergency. Existing optimizations for MMML-based affective computing services suffer from several shortcomings and limitations. State-of-the-art multi-modal applications use selective sensing techniques such as data filtering and compression [142], context-aware sensing [129], and heterogeneity-aware sensing [117] to reduce the total input data volume. Other techniques target model optimization to reduce the compute intensity of sense-making algorithms [5]. This shows these applications address multi-modal sensing and sense-making independently [106, 19, 97].

Table 7.1: Summary of MMML-based solutions for affective computing services.

Related Works	**Selective Sensing**	**Noise Awareness**	**Network Awareness**	**App Flexibility**	**Platform Agnostic**
[75]	✗	✗	✗	✗	✗
[142, 133, 129, 117]	✓	✗	✗	✗	✗
[5]	✗	✓	✗	✗	✗
AMSER [112]	✓	✓	✗	✓	✗
IMSER	✓	✓✓	✓	✓	✓

In addition, existing MMML-based affective computing solutions lacks supporting selective sensing together with noise-awareness, to inspect a modality and decide whether it provides insightful information for the prediction tasks. On the other hand, system environment dynamic significantly affect the sensing and sense-making configuration. For example, the network condition sensors are connected to the edge, needs to be considered to be able to evaluate the influence of data transfer speed and energy consumption on optimal sensing and sense-making configurations. In the following, we illustrate the resilience challenges in MMML-based affective computing services, effect of some environment dynamics on performance and the significance of intelligent and adaptive sensing and control in addressing those challenges through a motivational example of a pain monitoring application [195].

7.2.1 Sensory Data Variation

To emphasize the importance of resiliency challenges in MMML-based affective computing services and the significance of adaptive sensing and control in addressing those challenges, and for clarification, let me show a motivational example of a pain monitoring application [195]. The pain monitoring application acquires physiological data from different modalities viz., Electromyography (EMG), Electrocardiography (ECG), and Electrodermal Activity

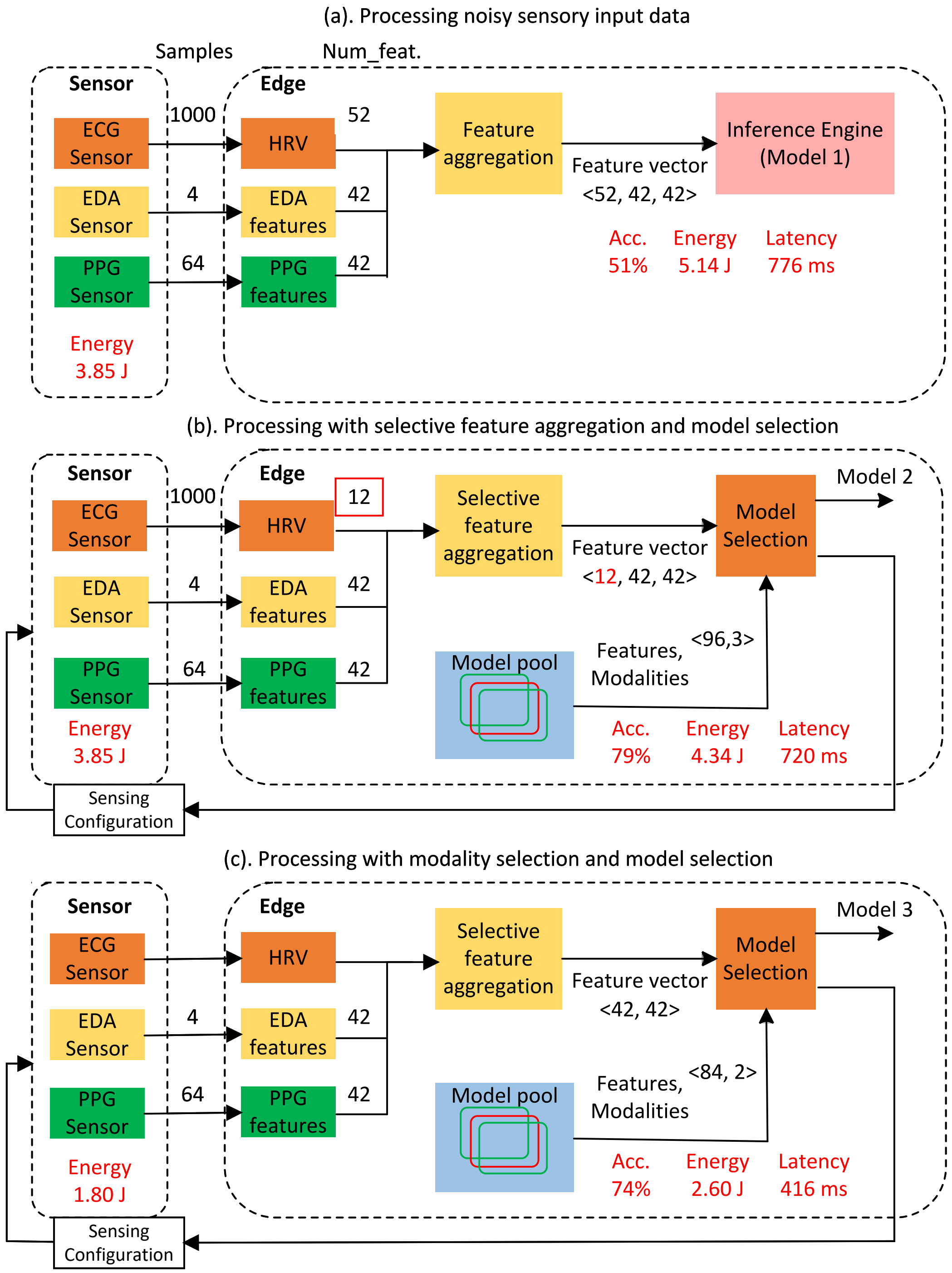


Figure 7.2: Example experimental scenario demonstrating adaptive sensing configuration for feature and modality selection.

(EDA) sensors to capture the autonomic nervous system activity against pain. Figure 7.2 shows the pipeline composed of multi-modal sensor data acquisition, feature aggregation, and machine learning model selection - for training and prediction. In this example, sensors from each of the ECG, EDA, and PPG modalities are sampled at 500Hz (2 channels), 4Hz, and 64 Hz, respectively. We extract relevant features from the raw input data and aggregate the features from all the modalities into an early-fused feature vector. We select a suitable machine learning model and train using the aggregated features to predict the pain levels. The model yields an average accuracy of 81% considering no noise on the input modalities. In practical scenarios, sensory data from one or more modalities can be noisy, and/or have data unavailability with perturbations such as motion artifacts, physical damages and battery shutdown [77]. In this example, we describe the implications of such sensory data perturbations on prediction accuracy and energy consumption, using different approaches of application orchestration.

(a) Processing noisy data. Figure 7.2 (a) shows the baseline scenario of processing sensory data without considering the noisy components. The ECG, EDA, and PPG modalities generate data of 8KB, 32B, and 512B, with 52, 42, and 42 features, respectively. In this case, the data from the ECG modality has a significant noisy component (e.g., due to improper patch contact to the chest), while the EDA and PPG modalities (e.g., collected from the wrist) generate quality inputs.

Ignoring the input quality, features from all the modalities are fused into a vector comprising 136 features. The model pool comprises a set of machine learning models suitable for different feature vectors and input modalities. From the model pool, a machine learning model that suits the feature vector fused from 3 modalities (3-modal, 136 feature) is chosen for training and prediction. This model yields an average accuracy of 51%, with an end-to-end latency of 776 ms, and energy consumption of 5.14 J and 3.85 J, for the edge and the sensor devices, respectively. The loss in prediction accuracy can be attributed to the noisy data from the ECG modality (garbage-in garbage-out effect). Further, this levies an unnecessary energy consumption incurred in sensing and processing the noisy data.

(b) Processing with selective feature aggregation. Figure 7.2 (b) shows an optimized scenario with selective feature aggregation for handling modalities with noisy data. By considering the noisy components and aggregating only a selected set of features from ECG modality, we improve prediction accuracy, while lowering the computational effort and energy consumption. In this scenario, the *Selective feature aggregation* selects 12 features from ECG modality (shown in red box) among the original 52 full scale features, and aggregates them with EDA and PPG features. The updated feature vector is sent to the *Model Selection* to choose a machine learning model that suits the feature vector for training and prediction. This model with selective feature aggregation yields an average accuracy of 79%, which is 35% higher than that of the baseline scenario with no input data considerations and only 2% less than the baseline with the no-noisy modality condition (ideal condition). Reduced features from the noisy ECG modality lowers the computational effort and energy consumption. In this case, selective feature aggregation improved the latency by 7% and energy consumption of the edge device by 16%. The sensing energy consumption does not change in this scenario since sensing configuration is unchanged.

(c) Processing with modality selection: Figure 7.2 (c) shows the optimized scenario with modality selection for handling modalities with unavailable data. In this scenario, data from the ECG modality is completely unavailable. With data from the EDA and PPG modalities being available, features from these modalities can be aggregated to build an updated feature vector. The *Selective feature aggregation* aggregates 42 features from EDA and 42 features from PPG modalities. From the *Models Pool*, the *Model Selection* chooses a machine learning model that suits the updated feature vector with reduced number of modalities and features (2-modal, 84 feature) for training and prediction. This model yields an average accuracy of 74%, which is 31% higher than that of the baseline scenario with no input data considerations.

Note that reducing the number of modalities to 2 (from 3 in Scenario (b)) lowers the accuracy only by 7% (compared to the ideal scenario). Dropping the unavailable ECG modality reduces the total data volume to be processed significantly. If the information on the sense-making decision is fed back to the sensing layer to adaptively change the sensing configuration, further energy saving at the sensing layer is also possible, which is critical in battery-powered wearable sensors. Overall, this leads to 46% improved latency, 50% lower energy consumption at the edge device, and 53% lower energy consumption at the sensors as compared to the baseline scenario of processing noisy data.

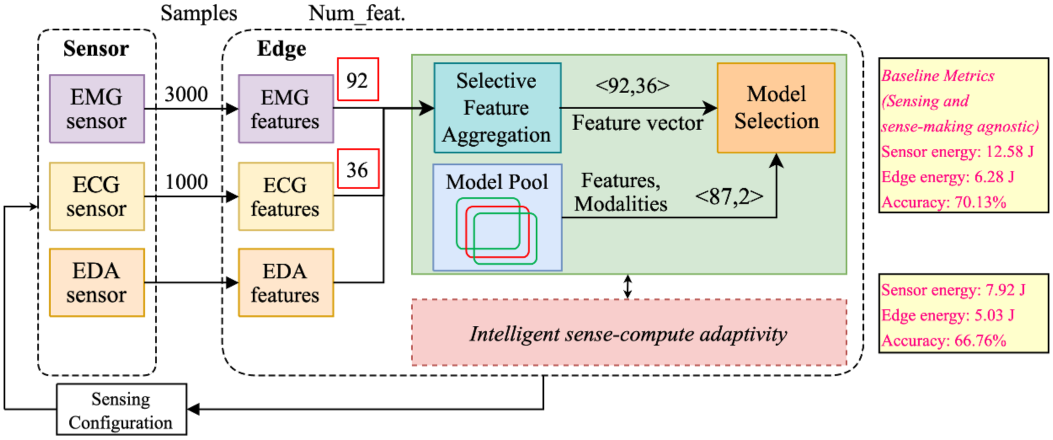


Figure 7.3: Processing with modality selection (selective feature aggregation and model selection) and additional intelligence for adaptive sense-compute

The example scenarios presented in Figure 7.2 demonstrates the advantages of monitoring input modalities for selective feature aggregation, modality selection, and model selection. However, this requires continuous monitoring of input modalities, and intelligent control for selective aggregation, modality, and model selection. Figure 7.3 shows a complete pipeline of a multi-modal processing, data acquisition from multiple sensors, feature extraction, selective feature aggregation, ML model selection. In this example, EMG, ECG, and EDA sensors are sampled at 500 (6 channels), 500 (2 channels), and 4 Hz (1 channel), respectively. Relevant features from each raw input modality is extracted and using an early fusion technique a single vector of features from all modalities is created. We then predict pain levels using a suitable machine learning models. In practical scenarios, one or more modalities can be distorted due to noises caused by motion artifacts, physical damages, battery shutdown [77]. In this specific example, EDA modality data contains motion artifact type of noise. This figure illustrates the implications of such perturbations on prediction accuracy and energy consumption of sensing and sense-making. The sensing and sense-making agnostic models yield baseline metrics of sensor energy, edge energy, and accuracy of 12.58 J, 6.28 J, and 70.13%, respectively. However, the addition of an intelligent sense-compute adaptivity to the system will result decent improvements on all the performance metrics.

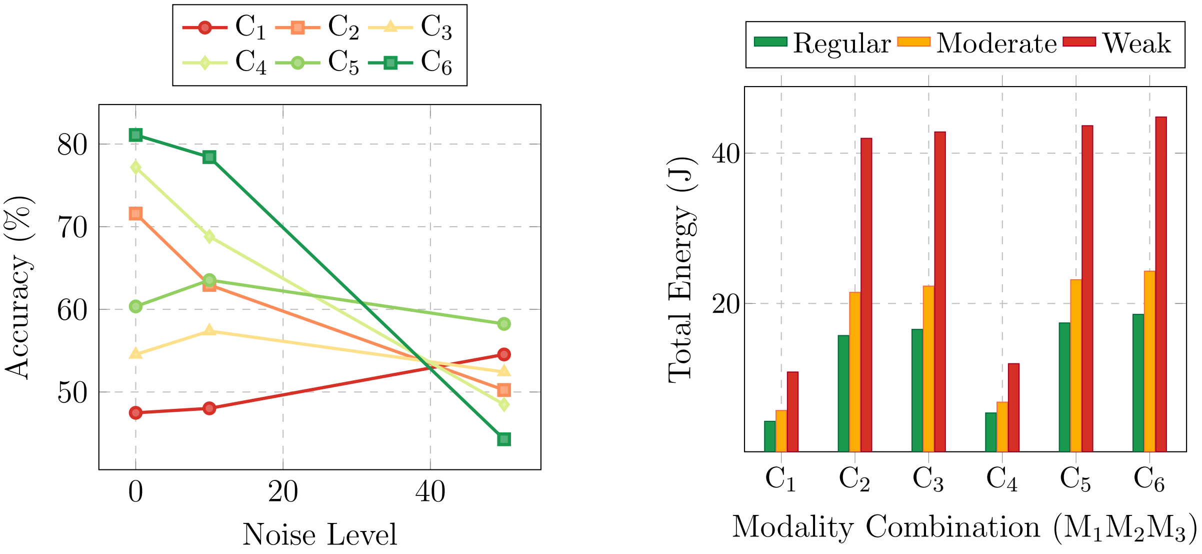
7.2.2 Environment Dynamics

In general, runtime dynamics of the system include connectivity, sensory input data quality, strength of the network, mobility and interaction of a given user to name a few. Such exploration is essential at design-time to achieve optimal configuration for stationary conditions. In reality, an IoT system's behavior dramatically changes over time due to variations in environment and system parameters.

Accuracy We demonstrate the impact of varying MMML inference on performance under different noise levels (NLs). We select between six configuration points with an accuracy between 48.2% and 81.7%. Figure 7.4b shows the accuracy achieved with varying portion of noise. With a single modality, the behavior of the system is different as the noise increases. The accuracy of the inference model when 35% of only one modality is available shows a positive correlation with the noise level, whereas the accuracy when full set of features for only one modality is available is not correlated with the noise level. Moreover, the accuracy of the MMML inference is in positive correlation with number of modalities and feature volume when noise is not available. However, as the noise level gets increased the MMML inference accuracy will most likely decrease once the amount of informative features gets corrupted with the noise.

Label	Modality		
	M_1	M_2	M_3
C_1	1	0	0
C_2	3	0	0
C_3	1	1	0
C_4	3	3	0
C_5	1	1	1
C_6	3	3	3

(a) Modality Combinations



(b) Accuracy for various noise levels

(c) Total Energy for various networks

Figure 7.4: Example experimental scenario design points demonstrating effect of (b) noise level on accuracy, (c) network on Total Energy

Network We consider three possible levels of network connections: (i) WiFi connectivity as a low-latency (regular) network that has the signal strength for better connectivity, (ii) 4G as a mid-latency (decent) network, and (iii) 3G as a high-latency (weak) network that has a weaker signal with poor connectivity. Figure 7.4c shows the total energy consumed for the pain monitoring application in different multi-modal volume configurations with all three networks. With a regular network, the energy consumed is lower when 35% of a single modality is available. The energy consumption increases as the reliable modalities are increasing. With a weak network, the energy consumption of the pain monitoring application is higher, as the poor signal strength adds more communication energy. Performance of the MMML inference is independent of the network connection, resulting in lowest response time. This demonstrates the spectrum of energy consumption with different set of reliable modalities, under varying network constraints.

7.2.3 Intelligent Orchestration

Runtime dynamics of the system in addition to requirements and opportunities affect orchestration strategies significantly. Sources of runtime variation across the system stack include connectivity, sensory input data quality, and strength of the network to name a few. Identifying optimal orchestration considering the sensing and sense-making opportunities and requirements in the face of varying system dynamics is a challenging problem. Making the optimal orchestration choice considering these varying dynamics is an NP-hard problem, while brute force search of a large configuration space is impractical for real-time applications. Understanding the requirements at each level of the system stack and translating them into measurable metrics enables appropriate orchestration decision making. Heuristic, rule-based, and closed-loop feedback control solutions are not efficient until reaching convergence, which requires long periods of time [175]. To address these limitations, RL approaches have been adapted for the joint-optimization decision making [158]. The model-free RL techniques operate with no assumptions about the system's dynamic or consequences of actions required to learn a policy. In other words, Model-free RL builds the policy model based on data collected through trial-and-error learning over epochs [175]. It enables *Platform agnostic* and *Application flexible* feature for RL-based sensing and sense-making frameworks where the agent finds optimal configuration through trial-and-errors during training phase with no assumption on *Platform* and *Application.*

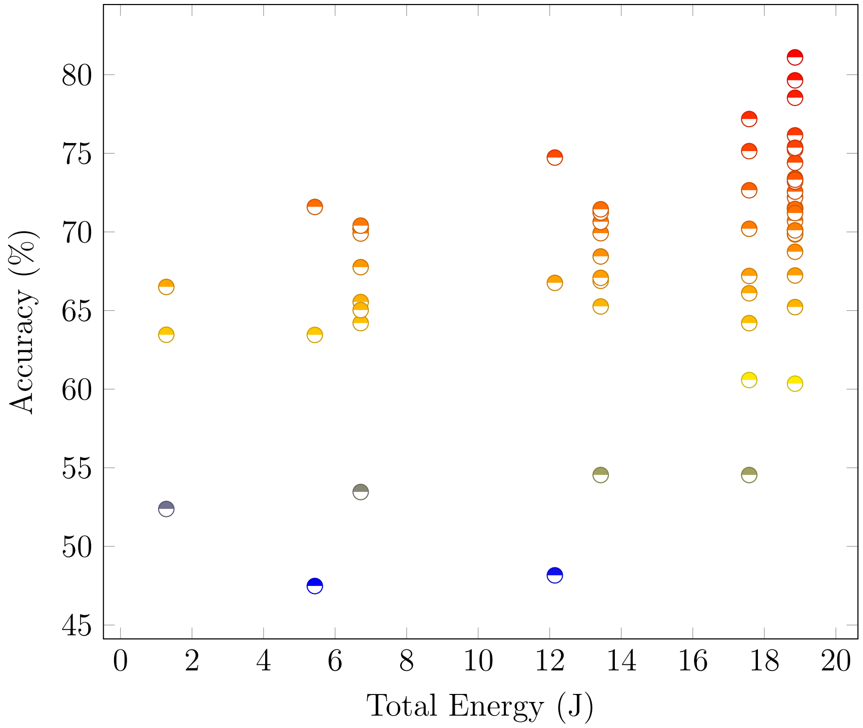


Figure 7.5: Design Space Configuration data points

7.2.4 Contributions

The ideal sensing and sense-making configuration provides maximum inference accuracy and minimum energy consumption. The literature lacks a comprehensive solution which is *noise-aware*, *network-aware*, *application flexible* and *platform agnostic* to run a variety of affective computing services, in particular on multipurpose wearables. The motivational example scenarios presented in Figures 7.2, 7.3, 7.4, and 7.5 demonstrate the advantages of monitoring input modalities for selective feature aggregation, modality selection, and model selection in presence of dynamics of the environment the importance of an intelligent orchestration. However, this requires intelligent sense-compute adaptivity by continuous monitoring of input modalities, and intelligent control for selective aggregation, modality, and model selection. To this end, we propose an automated framework for intelligent multi-modal sensing to improve resilience and energy efficiency of MMML-based affective computing services. Our solution presents a joint sensing and sense-making co-optimization using an RL-agent as a general solution for affective computing services considering energy consumption while meeting accuracy requirements. Table 7.1 summarizes and positions our contributions with respect to state-of-the-art.

7.3 Adaptive Multimodal Sensing and Sense-Making Framework (AMSER)

We deploy a commonly used affective computing services architecture to implement our AMSER approach, that consists of multiple sensor devices at the sensor layer and computing resources at the edge layer. Figure 7.6 shows an overview of the proposed framework. The *sensor layer* includes the multi-modal signal capability needed for affective computing monitoring applications. The edge layer is a computing device providing data gateway, data processing, signal monitoring, adaptive control, and inference at the edge.

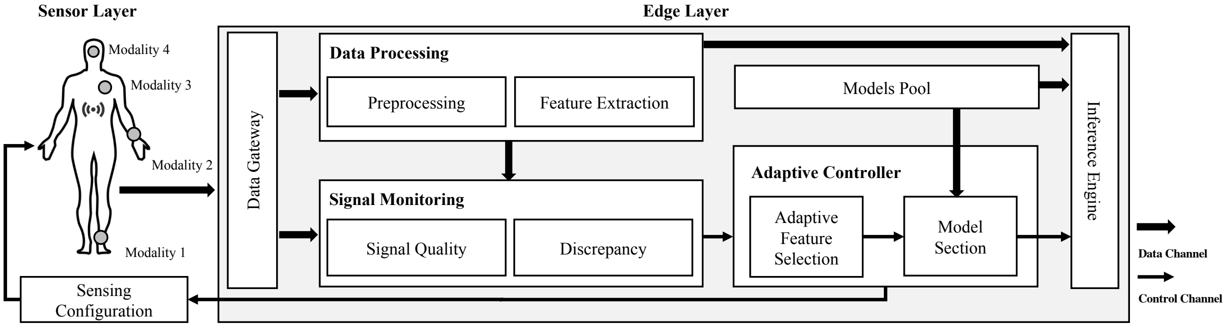


Figure 7.6: System Architecture Overview.

We detail each level of the framework below:

Data Gateway receives raw signal data from multi-modal sensors. The physiological signals can potentially contain two main types of noises: Baseline Wander (BW) and Motion Artifacts (MA).

Data Processing performs pre-processing and feature extraction with the data collected from multi-modal sensors.

Pre-processing: module consists of data synchronization of multi-modal signals, and bio-signal processing (e.g., peak detection) from different modalities such as ECG, PPG, and EDA that are essential for feature extraction, and finally filtering sensor modality inputs to produce clean signals.

Feature Extraction: module extracts informative and non-redundant values from the filtered signals. We extract time domain (mean, standard deviation, rms) and frequency domain (power spectral density, median frequency, central frequency) handcrafted features among the automatic features using a variational autoencoder for all modalities [10]. This process facilitates subsequent learning, leading to better interpretations for signal quality assessments.

Signal Monitoring handles quality assurance of the data and features to be used in the inference engine. This module observes the system parameters, disruptive events, data quality, and control flow to analyze the situation and context. *Signal Monitoring* assesses data and its extracted features quality by monitoring key parameters from the sensing phase to identify events and triggers for joint optimization of sensing and sense-making. For example, a disruptive event in input data of a specific modality (e.g., motion artifacts or sensor detachment) is a trigger at the sensing phase. In this regard, as shown in Algorithm 1, signal monitoring assesses data (D_M^i) and its extracted feature (F_M^i) quality and outputs the level of reliability of each modality (Rel_M^i).

Signal quality: We consider three levels of signal quality viz., reliable, noisy, and uncertain. *Signal quality* will assess the quality of sensing in terms of electrodes attachment, signal-to-noise ratio (SNR), and motion artifacts to determine the level of reliability. A signal with no noisy components is labeled as *reliable*. Signals extracted from sensor modalities whose electrodes are detached from the subject's body and signals with an SNR lower than a fixed threshold [116] are labeled as *noisy*. Signals with an SNR above the fixed threshold yet below the acceptable levels are labeled as *uncertain*.

Discrepancy Detector: The *Discrepancy Detector* acts as a second layer for assessing the quality of signals that are labeled as *reliable* in the *Signal Quality* phase by analyzing modality-specific parameters to identify any discrepancies. *Discrepancy Detector* will recognize confounding factors in each modality to identify levels of signal reliability. For instance, for ECG and PPG modalities, we use the range of HR, RR interval (RRi) length, the ratio of Max RRi to min RRi with their thresholds as the parameters to determine discrepancies [116]. As another example, for EDA modality, we use low amplitude and very low and steady tonic level for more than 1 minute as the discrepancy detecting parameter [21, 132].

Algorithm 1 Signal Monitoring Strategy.

```
$Rel^i_M \leftarrow RELIABLE$
if $sensorStatus^i_M$ == DETACHED then
    $Rel^i_M \leftarrow NOISY$
else
    if SNR $\leq$ threshold1 then
        $Rel^i_M \leftarrow NOISY$
    else
        if threshold1 $\leq$ SNR $\leq$ threshold2 then
            $Rel^i_M \leftarrow UNCERTAIN$
        end if
    end if
end if
     for each modality $D^i_M$
if Rules($D^i_M$) == PASS then
    $Rel^i_M \leftarrow RELIABLE$
else
    $Rel^i_M \leftarrow UNCERTAIN$
end if
```

Adaptive Controller analyzes inputs from the signal monitoring module to adaptively select features and modalities, and configure sensing and computation models. We store pre-trained models for different combinations of signal modalities and aggregated feature vectors in a pool of models. The adaptive controller is designed to holistically and dynamically control the quality of sensing and sense-making, as shown in Algorithm 2. The adaptive controller considers the reliability of each sensor modality (Rel^i_M), the feature vector (F^i_M) extracted from *Signal Monitoring*, and the model pool for feature aggregation and model selection. The controller module adaptively selects the features that are relevant for prediction accuracy, and a learning model that is suitable for the updated feature vector using a simple rule based approach. In case a modality is labelled as *uncertain*, we process the modality by dropping some of the less prominent features. If a modality is *noisy*, we drop the input data from the modality entirely. The controller will generate a signal to update the sensing configuration, such that *noisy* sensor modalities can be turned off. Sensor modalities are turned into the idle mode when the noise cannot be mitigated and thereby eliminating the unnecessary computation and communication penalty. Finally, *Model Selection* utilizes the selected features, and the current models available in the *Model Pool* to load a preferable model for the *Inference Engine* taking into consideration the sensing status and situation. Algorithm 2 is a simple proof-of-concept rule-based policy to show the potentials of this approach. More advanced algorithms (e.g., using reinforcement learning or fuzzy control) can be devised to make intelligent decisions.

Algorithm 2 Adaptive Controller.

```
$Sens_M^i \leftarrow ON$
if $Rel_M^i == NOISY$ then
    $Sens_M^i \leftarrow OFF$
else
    if $Rel_M^i == UNCERTAIN$ then
        $F_M^i \leftarrow$ Choose subset of $F_M^i$
    else
        $F_M^i \leftarrow F_M^i$
    end if
end if
if $Sens_M^i$ and $F_M^i$ then
    $Inf_M^i \leftarrow MP_M^i$
end if
```

7.4 Evaluation of AMSER

We evaluate the proposed adaptive sensing framework through two case studies of pain assessment [79, 8, 30] and stress monitoring [155]. Both case study applications require continuous monitoring of multiple modalities including ECG, EMG, PPG, and EDA signals, to detect pain and stress levels. Our evaluation platform comprises a sensory node - to collect physiological signals of ECG, EMG, PPG, and EDA from the subject, and an edge node - to execute the inference. We implement the proposed method on a ODROID-XU3 with an octa-core Exynos processor as the edge device. In the following, we detail the two case studies and evaluate the case study applications' accuracy, energy efficiency, and performance using our AMSER proposed framework through different scenarios with different types and levels of noisy and unreliable input modalities. Further, we compare our AMSER approach against state-of-the-art multi-modal pain monitoring application that use classification of physiological signals [74] as the baseline ideal scenario.

7.4.1 Scenarios

We validate and demonstrate the efficacy of our AMSER approach through the case study of pain assessment from the affective computing domain. We evaluated our proposed AMSER approach in 4 different scenarios with different noise component levels for each case study. Table 7.2 summarizes different scenarios (S1-S4), showing the type of noise induced in each scenario. Scenario 1 is the ideal baseline case with no noise components, Scenario 2 has *uncertain* signal components, Scenario 3 has one modality with *noisy* component, and Scenario 4 has two *noisy* modalities. In Scenario 1, there is no noise component, and thus no modality selection or feature selection is applied. Scenario 2 comprises of baseline wandering noise in addition to the original data for all the modalities. With the presence of noise, our proposed AMSER approach selects specific significant features, reducing the total number of features from the original feature vector. In Scenario 3, we add additional motion artifacts on top of one modality while all the modalities still have a baseline wander noise. In this scenario, the noisy modality is completely unreliable. Thus, our proposed method will drop the modality and uses a different model from *Model Pool* with 3 and 2 modalities for pain and stress case studies, respectively. Scenario 4 comprises 2 modalities facing severe motion artifact noises. In case of pain assessment, ECG and EMG modalities are noisy with entirely unreliable data. Our proposed adaptive sensing will drop both the ECG and EMG modalities and select a learning model with 2 modalities for pain assessment application. In case of stress monitoring, ECG and EDA modalities are noisy. Our proposed adaptive controller drops both the ECG and EDA modalities and selects a learning model with the single PPG modality for stress monitoring.

Table 7.2: Experimental Scenarios for Pain and Stress applications. $+BW$, and $+MA$ represent the presence of baseline wandering and motion artifact noise, respectively.

	ECG	**EMG**	**PPG**	**EDA**
S1	-	-	-	-
S2	+BW	+BW	+BW	+BW
S3	+BW	+BW	+BW+MA	+BW
S4	+BW+MA	+BW+MA	+BW	+BW

We evaluate the accuracy and energy efficiency of both the pain monitoring application (which estimates the pain intensity using 4 physiological signals ECG, EMG, EDA, and PPG), and the stress monitoring application (which calculates the stress level using 3 physiological signals ECG, PPG, and EDA) using the iHurt_DB [78]. We trained different models for each levels of pain intensities, varying in the types of modalities combined for the decision-making system. After pre-processing, we extract a set of unique features from each modality viz., 52 features of ECG, 42 features of EDA, and 42 features of PPG [79, 8, 30].

Table 7.3: Decision made by AMSER during each scenario.

	ECG	**EMG**	**PPG**	**EDA**	**Feature Vol (%)**
S1	✓	✓	✓	✓	100
S2	✓	✓	✓	✓	63.8
S3	✓	×	✓	✓	35.8
S4	×	×	✓	✓	31.3

7.4.2 Accuracy Evaluation

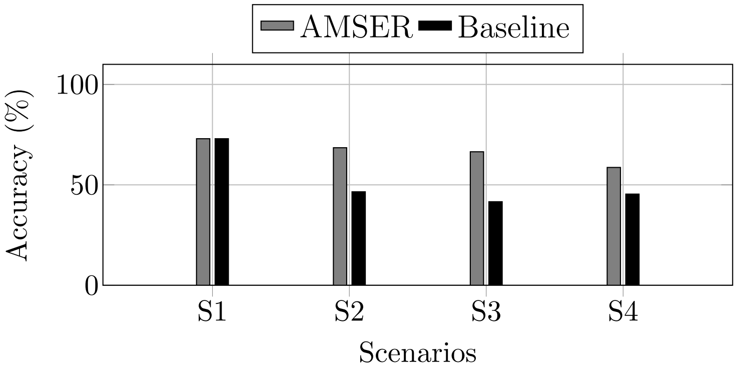


Figure 7.7: Accuracy analysis for AMSER vs. Baseline [74] for Pain application.

We compare the accuracy achieved by the model in each scenario with noisy components (S2-S4) against the baseline ideal scenario $S1$ with no adaptive modality and feature selection. Figure 7.7 shows the accuracy (in %) for pain assessment and stress monitoring case studies under different scenarios. Our proposed adaptive multimodal sensing technique provides an improved accuracy in each of the scenarios for both the case studies, with a maximum gain of 22% in S2 for pain assessment. In the presence of a lower level of noise i.e., *uncertain* signal, our proposed AMSER approach still utilizes the *uncertain* modality with selected features, instead of either dropping or holding the entire feature sets of the noisy modality. This approach leads to a 22% and 17% of accuracy improvement for pain and stress applications respectively, in comparison with the baseline model which uses all the features from the noisy modality. With a higher level noise i.e., *noisy* modality in scenario $S3$, our proposed AMSER approach leads to 15% and 10% accuracy improvement for pain and stress applications, respectively, compared to the baseline which uses all the modalities. Lastly, scenario $S4$ shows that dropping 2 noisy modalities results in better accuracy (13% and 11% improvement for pain and stress applications, respectively), as compared to the baseline using noisy modalities.

7.4.3 Efficiency and Performance Evaluation

We evaluate the efficiency and performance of AMSER in comparison with the baseline. Figure 7.8 shows the energy efficiency improvement during four scenarios for edge and sensor devices. We report the results for both Pain and Stress applications. Figure 7.8 (a) shows the energy efficiency of the pain assessment application. In Scenario $S4$, the energy gain is 2.19× and 5.63× higher than the baseline for the edge and sensors, respectively. In this case, AMSER provides 13.27% better accuracy by dropping two noisy modalities (See Figure 7.7 and Table 7.3). Figure 7.8 (b) shows the energy efficiency for the stress monitoring application. In Scenario $S3$, our evaluation shows that AMSER improves the energy efficiency of the stress monitoring application by 1.32× and 2.63× for the edge and sensor devices, respectively (See Figure 7.8 (b)). In this case, AMSER improves the accuracy and performance by 10.56% and 27%, respectively (See Figure 7.7). Through the Scenario $S3$ for Stress application, AMSER keeps 31.6% of the features and drops the unreliable features (See Table 7.3). Figure 7.9 (a) shows the performance gains at the edge device for the pain assessment and stress monitoring applications using our proposed AMSER approach, as compared to the baseline Scenario $S1$. In Scenarios $S2$-$S4$, our proposed AMSER approach reduces the number of features and/or drops input data from *noisy* modalities. This lowers the total computational effort, leading to an increase in performance, as compared to the baseline. Figure 7.9 (b) shows the data volume transferred between the sensory devices and edge node using our proposed AMSER approach, in comparison with the baseline. Our AMSER approach adaptively selects features and/or drops *noisy* modalities, which reduces the total input data volume significantly. For instance, in Scenario $S4$, two noisy modalities are entirely dropped, leading to more than 6× reduction in transmitted data volume. Such reduction in data volume to be transmitted further reduces the communication latency and energy expense incurred in transmitting noisy data.

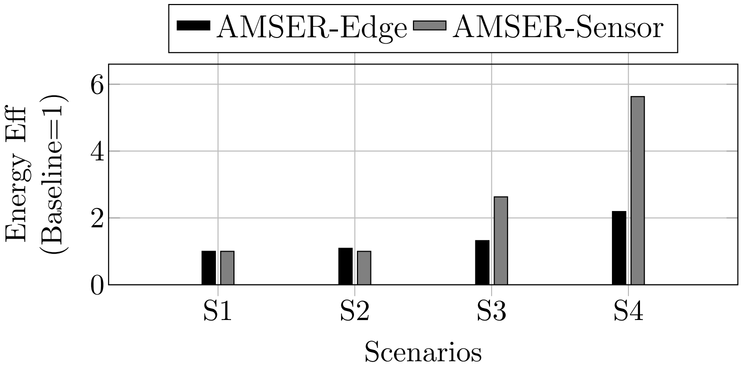
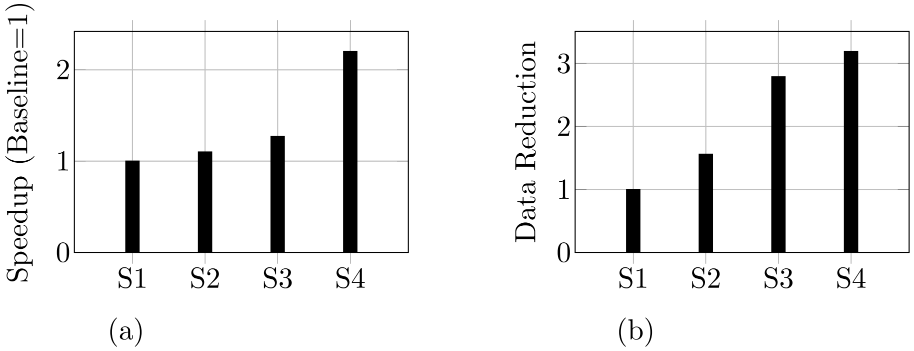


Figure 7.8: Energy efficiency analysis of the edge and sensor device for AMSER vs. Baseline [74] for Pain application.



(a)

(b)

Figure 7.9: (a) Performance analysis of the edge device for AMSER vs. Baseline [74] for Pain application. (b) Data volume transferred between the sensors and edge for AMSER vs. Baseline.

7.5 Intelligent Multimodal Sensing and Sense-Making Framework (IMSER)

We deploy a commonly used affective computing system architecture to implement our IMSER approach, that consists of multiple sensor devices at the sensor layer and computing resources at the edge layer. Figure 7.10 shows an overview of the proposed framework. The *sensor layer* includes the multi-modal signal capability needed for affective computing services. The edge layer is a computing device providing data gateway, data processing, quality monitoring, intelligent and adaptive control over feature and model selection, and inference at the edge.

We detail each level of the framework below:

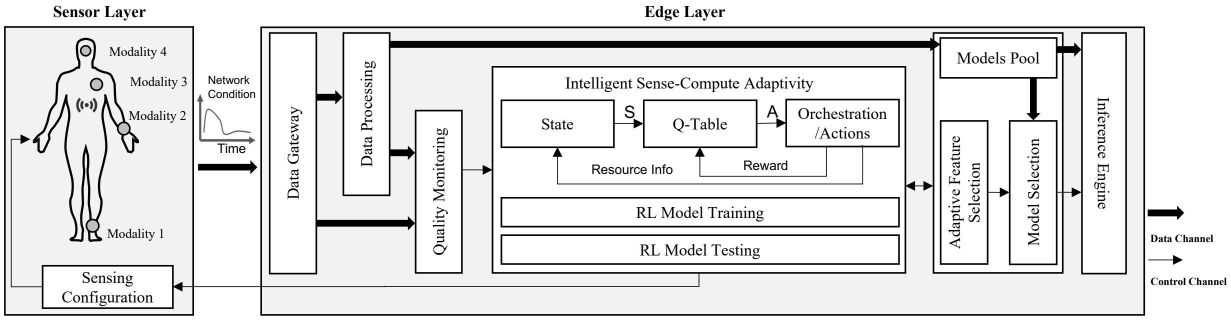


Figure 7.10: System Architecture Overview.

Data Gateway receives multi-modal raw input data from multiple sensors. The physiological signals can potentially contain noise in different levels and portions of the data window segment from zero (noise-free) to 100% (super noisy).

Data Processing performs pre-processing and feature extraction using the data collected from multi-modal sensors. *Preprocessing* module consists of synchronization of multi-modal signals and labels, filtering sensor modality inputs to produce clean signals, and additional parts of the affective signal processing (ASP) pipeline (e.g., peak detection, normalization) based on the modality such as EMG, ECG, and EDA and need for affective computing services such as pain monitoring and stress monitoring that are essential for feature extraction. *Feature Extraction* module extracts insightful and informative values from the preprocessed signals. We extract handcrafted features in time domain (mean, standard deviation, rms) and frequency domain (e.g. power spectral density, median frequency, central frequency) among the automatic features using a variational autoencoder for all modalities [10]. This process facilitates subsequent learning, leading to better interpretations for quality assessment module.

Quality Assessment handles quality assurance of the data and features to be used in the inference engine. This module observes the system parameters, disruptive events, data quality, and control flow to analyze the situation and context. *Quality Assessment* module assesses signal and its extracted features quality by monitoring key parameters from the sensing phase to identify events and triggers for joint optimization of sensing and sense-making. For example, a trigger at the sensing phase could be any disruptive event in input data of a specific modality (e.g., motion artifacts or sensor detachment).

Signal Quality: will assess the quality of sensing in terms of electrodes attachment, signal-to-noise ratio (SNR), and motion artifacts to determine the level of reliability of the signal. We consider three levels of signal quality viz., reliable, noisy, and uncertain. A signal with no noisy components is labeled as *reliable*. Signals extracted from sensor modalities whose electrodes are detached from the subject's body and signals with an SNR lower than a fixed threshold [116] are labeled as *noisy*. Signals with an SNR above the fixed threshold yet below the acceptable levels are labeled as *uncertain*.

Feature Quality: acts as a second layer for assessing the quality of signals that are labeled as *reliable* in the *Signal Quality* sub-module by analyzing modality-specific parameters to identify any discrepancies. *Feature Quality* will recognize confounding factors in each modality to identify levels of signal reliability.

Intelligent Sense-Compute Adaptivity analyzes inputs from the quality assessment module to intelligently communicate with the adaptive controller to select features and modalities, and configure sensing and computation models. We store pre-trained models for different combinations of signal modalities and aggregated feature vectors in a pool of models. The *Intelligent Sense-Compute Adaptivity* is designed to holistically and dynamically control the quality of sensing and sense-making using a reinforcement learning agent, as described in the following.

7.5.1 Reinforcement Learning Agent

Reinforcement learning (RL) is widely used to automate intelligent decision making based on experience. Information collected over time is processed to formulate a policy which is based on a set of rules. Each rule consists of three major components viz., (a) state, (b) action, and (c) reward. Among the various RL algorithms [175], Q-learning has low execution overhead, which makes it a perfect candidate for runtime invocation. Figure 7.10 depicts high-level block diagram for our agent. The RL agent is invoked at runtime for intelligent orchestration decisions. Our agent is composed as follows:

State Space: Our state vector is composed of Modality availability, Feature volume per each modality, Network condition, and noise level. Table shows the discrete values for each component of the state. Modality availability (represented as MA) is a binary value that states our framework either collects sensory data for that modality or is idle. Feature volume (FM) states what percentage of the feature volume for each modality is incorporated for the further analysis after data processing. FM is considered as discrete value between 0 to 100%. Network condition (represented as NC) shows how the sensors are connected to the edge device. We consider three different network connections which demonstrates different data transfer speed and power consumption. Noise level (represented as NL) states what percentage of each modality is noisy. The state vector at time step τ is defined as follows:

$$S_\tau = \{FM_1, FM_2, FM_3, MA_1, MA_2, MA_3, NL, NC\} \tag{7.1}$$

Action Space: The action vector consists of increase/decrease feature volume for each modality. In other words, the agent takes an action to change feature volume of only one of modalities at each time step. The action vector at time step τ is defined as follows:

$$A_\tau = \{A_1, A_2, A_3\} \tag{7.2}$$

Table 7.4: State Discrete Values

State	Discrete Values	Description
FM_i	0,35%,70%,100%	Modality Feature Volume
MA_i	0,1	Modality Availability
NL	0, 20%, 50%	Modality Noise Percentage
NC	Regular, Moderate, Weak	Network Condition

Reward Function: The reward function is defined as the negative total energy consumption including edge and sensor devices. In our case, the agent seeks to minimize the energy consumption. To ensure the agent minimizes the energy consumption while satisfying the accuracy constraint, the reward R is calculated as follows:

$$\begin{aligned} &\text{if } \overline{Accuracy} > \text{constraint:} \\ &\qquad R_\tau \leftarrow -Energy \\ &\text{else:} \\ &\qquad R_\tau \leftarrow -Max\ Energy \end{aligned} \tag{7.3}$$

To apply the accuracy constraint, the minimum possible reward is assigned when the accuracy threshold is violated. On the other hand, when the selected action satisfies the accuracy constraint, the reward is negative energy consumption in that time step. Algorithm 3 defines our agent's logic with the epsilon-greedy Q-Learning:

Line Description

3: First the agent determines the current system state from the resource monitors.

4-8: Either the state-action pair with the highest Q-value is identified to choose the next action to take, or a random action is selected with probability ϵ.

9-10: The selected action is applied and normal execution resumes. The reward R_τ for the

Algorithm 3 Q-Learning Algorithm

```
1: while system is on do
2:     From Resource Monitoring:
           $S_\tau \leftarrow$ State at step $\tau$
3:     if $RAND < \epsilon$ then
4:         Choose random action $A_\tau$
5:     else
6:         Choose action $A_\tau$ with largest $Q(S_\tau, A_\tau)$
7:     end if
8:     Monitor total energy consumption
9:     Calculate reward $R_\tau$
10:    From Resource Monitoring:
           $S_{\tau+1} \leftarrow$ State at step $\tau + 1$
11:    Choose action $A_{\tau+1}$ with the largest $Q(S_{\tau+1}, A_{\tau+1})$
12:    To Updating Qtable:
           $Q(S_\tau, A_\tau) \leftarrow Q(S_\tau, A_\tau) + \alpha[R_\tau + \gamma Q(S_{\tau+1}, A_{\tau+1}) - Q(S_\tau, A_\tau)]$
13:    $S_\tau \leftarrow S_{\tau+1}$
14: end while
```

execution period is calculated based on measured consumed energy.

11-12: Based on the resource monitors, the new state $A_{\tau+1}$ is identified, along with the state-action pair with highest Q-value.

13: The Q-value of the previous state-action pair is updated.

14: The current state is updated, and the loop continues.

7.6 Evaluation of IMSER

We evaluate the proposed intelligent sense-computing framework through a case study of pain assessment [79, 8, 30]. Continuous monitoring of multiple sensor modalities including ECG, EMG, PPG, and EDA signals is required for this case study application to detect pain levels. Our platform for evaluation comprises a sensory node - to collect physiological input modalities of ECG, EMG, PPG, and EDA from the subject, and an edge node - to control sensing and computation and execute the inference. The RL agent's goal is to minimize average response time while satisfying the accuracy and energy constraint. This enforces quality control by imposing a strict threshold on the average DL model accuracy. The proposed method is implemented on an ODROID-XU3 with an octa-core Exynos processor as the edge device. In the following, we detail the case study and evaluate the case study applications' accuracy, energy efficiency, and performance using our IMSER proposed framework through different scenarios with different levels of noisy input window segments and network strength for all modality. Further, we compare our IMSER approach against AMSER and state-of-the-art multi-modal pain monitoring application that use classification of physiological signals [74] as the baseline ideal scenario.

7.6.1 Hyper-parameter Tuning

An RL agent has a number of hyper-parameters that impact its effectiveness (e.g., learning rate, epsilon, discount factor, and decay rate). The ideal values of parameters depend on the problem complexity, which in our case scales with the number of modalities, noise level, and network condition. In order to determine the learning rate and discount factor, we evaluated values between 0 and 1 for each hyper-parameters. We observed that a higher learning rate converges faster to the optimal, meaning the more the reward is reflected to the Q-values, better the agent works. We also observed that a lower discount factor is better. This means that the consecutive actions have a weak relationship, so that giving less weight to the rewards in the near future improves the convergence time.

7.6.2 Scenarios

We validate and demonstrate the efficacy of our IMSER approach through the exemplar pain assessment application from the affective computing domain. In this work, we conduct experiments under seven unique scenarios with varying noise levels and network conditions

Table 7.5: Experimental Scenarios. Each scenario represent the noise level out of 100 for each modality and the network condition.

Scenarios		Noise Level	Network
S1	Scenario 1	0	Regular
S2	Scenario 2	20	Regular
S3	Scenario 3	20	Moderate
S4	Scenario 4	20	Weak
S5	Scenario 5	50	Regular
S6	Scenario 6	50	Moderate
S7	Scenario 7	50	Weak

for our case study. The experimental scenarios are summarized in Table 7.5, representing a combination of noise level among with a regular, moderate and weak network signal strength. The regular network has no transmission delay, while we add 10ms and 20ms delay to all outgoing packets to emulate the moderate and weak connection behavior, respectively. Putting together the noise level and the network condition creates unique experimental scenarios. Scenario 1 is the ideal baseline case with no noise and a regular network, Scenario 2 has signal components with 20% noise level among with a regular network condition. Scenario 3 and 4 are similar to scenario 2 in terms of noise level percentage but with moderate and weak network condition, respectively. Scenario 5 has modalities with 50% noise level with a regular network condition. Scenario 6 and 7 are similar to scenario 5 in terms of noise level percentage but with moderate and weak network condition, respectively. The proposed IMSER framework has the capability to choose the right configuration with a constraint on accuracy. We define Min and Max accuracy constraints empirically. We set Min to be the median of the accuracy of all the potential decisions. Within the presence of such constraint, IMSER will choose the configuration that provides an accuracy with the smallest number greater than Min. On the other hand, Max refers to the situation when there is no constraint on accuracy and IMSER will try to maximize the accuracy value. In Scenario 1, there is no noise component and network condition is regular, and thus no modality selection or feature selection is applied. Scenario 2 comprises of 20% level of noise in addition to the original data for all modalities within a regular network. In this scenario when there is a Min accuracy constraint, our proposed IMSER approach drops ECG modality and selects 66% of the features from EMG modality, reducing the total number of features from the original feature vector, while it will not drop any modality and select the most significant features from ECG modality when there is no constraint on accuracy. Scenario 3 and 4 demonstrates that network condition does not have any effect on prediction model accuracy. Scenario 5, comprises of 50% noise level added to the original data for all modalities within a regular network. In this scenario, when there is a Min accuracy constraint, the EDA modality is completely unreliable. Thus, IMSER will drop the EDA modality and select 33.3% of the features from other two modalities, reducing the total number of features from original feature vector, while it will not drop any feature or modality when there is no constraint on accuracy. Scenario 6 and 7 replicate the same scenario in a moderate and weak network condition, respectively.

We evaluate the accuracy and energy efficiency of the pain monitoring application (which estimates the pain intensity using 3 physiological signals ECG, EMG, and EDA) using the iHurt Pain DB [78]. We trained different models for each levels of pain intensities, varying in the level of noise, network condition and types of modalities combined for the decision-making system. After pre-processing, we extract a set of unique features from each modality viz., 52 features of ECG, 152 features of EMG, and 42 features of EDA [79, 8, 30].

7.6.3 Accuracy Evaluation

We compare the accuracy achieved by the model in each scenario with noisy components (S2-S7) against the baseline ideal scenario $S1$ with no intelligent modality and feature selection.

Table 7.6: Decision made by AMSER and IMSER during each scenario.

Method	Modality	Scenarios													
		S1		S2		S3		S4		S5		S6		S7	
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
AMSER	ECG	✓	✓	✓	✓	N/A	N/A	N/A	N/A	✓	✓	N/A	N/A	N/A	N/A
	EMG	✓	✓	✓	✓	N/A	N/A	N/A	N/A	✗	✗	N/A	N/A	N/A	N/A
	EDA	✓	✓	✓	✓	N/A	N/A	N/A	N/A	✓	✓	N/A	N/A	N/A	N/A
	Feat Vol	100%	100%	63.8%	63.8%	N/A	N/A	N/A	N/A	31.3%	31.3%	N/A	N/A	N/A	N/A
IMSER	ECG	✓	✓	✗	✓	✗	✓	✗	✓	✓	✓	✓	✓	✓	✓
	EMG	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	EDA	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✓	✗	✓
	Feat Vol	100%	100%	55.6%	77.7%	55.6%	77.7%	55.6%	77.7%	33.3%	55.6%	33.3%	55.6%	33.3%	55.6%

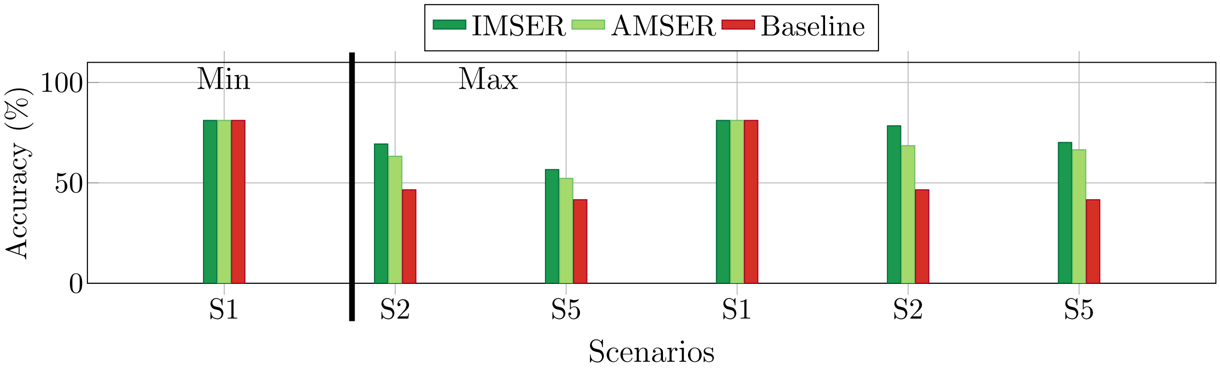


Figure 7.11: Accuracy analysis for IMSER vs. AMSER vs. Baseline [74] for Pain application.

Figure 7.11 shows the accuracy (in %) for pain assessment case study under different scenarios. Our proposed intelligent multimodal sense-compute technique provides an improved accuracy in each of the scenarios with a maximum gain of 23% and 32% in S2 with Min and Max accuracy constraint for pain assessment. In the presence of a lower level of noise i.e., 20%, our proposed IMSER approach drops ECG and utilizes the EMG with selected features and drops features from only ECG modality to meet the Min and Max accuracy constraints, instead of holding the entire feature sets of the noisy modalities. This approach leads to a 23% and 32% accuracy improvement in scenario $S2$ with Min and Max accuracy constraint for pain assessment application in comparison with the baseline model which uses all the features from the noisy modalities. With a higher level of noise i.e., 50%, in scenario $S5$, our proposed IMSER approach leads to 15% and 28% accuracy improvement for Min and Max accuracy constrains, respectively, compared to the baseline which uses all the modalities.

7.6.4 Efficiency and Performance Evaluation

We evaluate the efficiency and performance of IMSER in comparison with the baseline. Figure 7.12 shows the energy efficiency improvement during seven scenarios for edge and sensor devices. We report the results for Pain application with Min and Max accuracy constraint. In Scenario $S4$, the energy gain to meet Min accuracy constraint is 1.36× and 1.33× higher than the baseline for the edge and sensors, respectively. In this case, IMSER provides 22.82% better accuracy by dropping the noisy modality (ECG) and selecting informative feature from EMG modality (See Figure 7.11 and Table 7.6). Figure 7.13 (a) shows the performance gains at the edge device for the pain assessment application using our proposed IMSER approach, as compared to the baseline Scenario $S1$. In Scenarios $S2$-$S7$, our proposed IMSER approach reduces the number of features and/or drops input data from *noisy* modalities. This lowers the total computational effort, leading to an increase in performance, as compared to the baseline. Figure 7.13 (b) shows the data volume transferred between the sensory devices and edge node using our proposed IMSER approach, in comparison with the baseline. Our IMSER approach intelligently selects features and/or drops *noisy* modalities, which reduces the total input data volume significantly. For instance, in Scenario $S4$, one noisy modality is entirely dropped and another modality contribute with two third of its most informative features, leading to more than 6× reduction in transmitted data volume. Such reduction in data volume to be transmitted further reduces the communication latency and energy expense incurred in transmitting noisy data.

7.6.5 Overhead Analysis

Exploration Overhead: We evaluate the time required by the proposed agent for the training phase to identify an optimal policy. Figure 7.14 shows the training phase under different accuracy constraints using Q-Learning algorithm. This shows that when training a

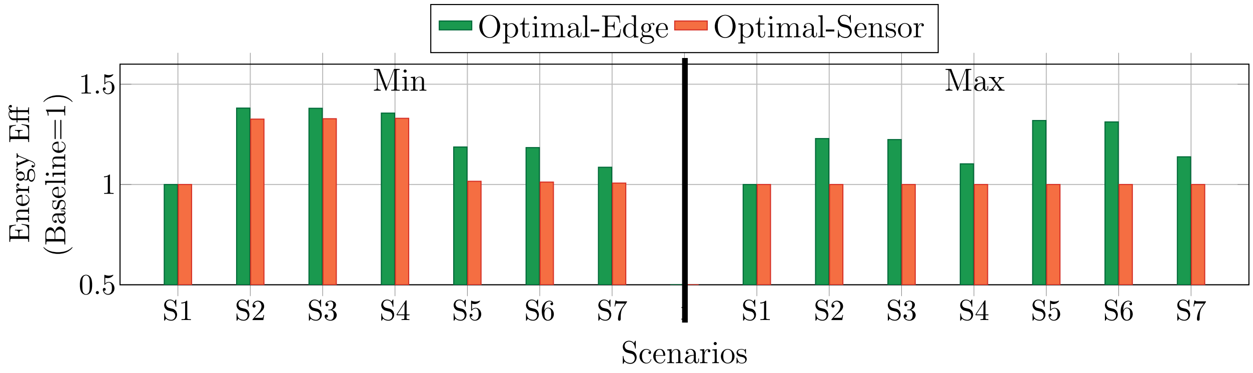


Figure 7.12: Energy efficiency analysis of the edge and sensor device for IMSER and AMSER vs. Baseline [74] for Pain application.

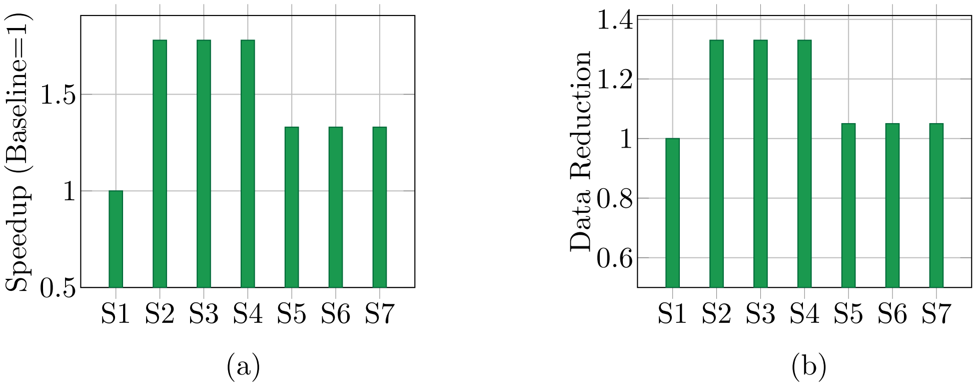


Figure 7.13: (a) Performance analysis of the edge device for IMSER vs. Baseline [74] for Pain application. (b) Data volume transferred between the sensors and edge for IMSER vs. Baseline.

model from scratch, the reward converges after about about 170 inference runs on average. However, increasing the accuracy constraint leads to a more complex problem and therefore increase in convergence time.

Runtime Overhead: To demonstrate the viability of mobile inference deployment, we evaluate the IMSER runtime overhead. The performance overhead of RL algorithm in IMSER is, on average, 20 μs for training, excluding the time for inference execution. It corresponds to 1.2% of the lowest inference latency. In addition, when using the trained Q-table, the overhead can be reduced to 7.3 μs with only 0.3% overhead. This result means it takes 18.1 μs to measure the inference results, calculate the reward, and update the Q-table. The

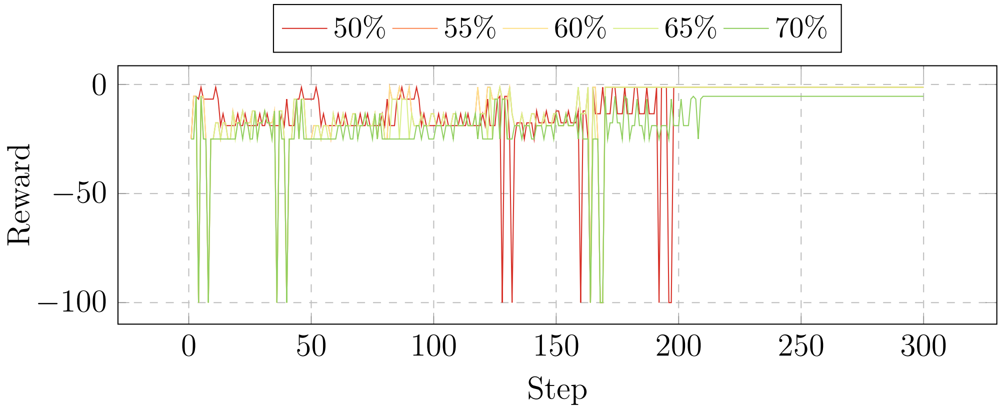


Figure 7.14: Training overhead for Q-Learning algorithm under different accuracy constraints

energy overhead is only 1% and 0.2% of the total system energy consumption, when training the Q-table and exploiting the trained Q-table, respectively.

Chapter 8

Conclusions

The aim of this thesis has been to develop an automatic and versatile objective pain assessment tool from the continuous monitoring of multiple sensing modalities to help advance the understanding, measurement of pain in real life. Moreover, a design space exploration of accuracy-performance-energy trade-offs and sense-compute co-optimization for multimodal machine learning methods was performed. In this concluding chapter, the findings, strength and limitations will first be presented, and then followed by the contributed highlights of the study and recommendations for future research.

8.1 Main findings

We presented a multimodal machine learning framework for classifying pain in real post-operative patients from the iHurt Pain Database. Both traditional handcrafted features and deep learning generated automatic features were extracted from physiological signals (ECG, EDA, EMG, PPG). We conducted several experiments to perform binary classification among four different pain intensities vs baseline levels of pain. Models for each of these intensities were varied based on the modalities, different types of augmentation techniques (Smote, Snorkel, or both), machine learning algorithms, and the type of modality fusion used. Our results showed that binary pain classification greatly benefits from using label augmentation techniques in conjunction with automatic features. The multimodal model outperformed the single modal models, with the exception of the last pain intensity. The BL vs PL4 model with the best results was trained on EMG data alone, which suggests that facial muscle activation can play a vital role in distinguishing higher pain intensities from baseline levels of pain. This is consistent from a clinical perspective because higher pain intensities are more commonly associated with acute pain.

ML-driven affective computing applications have different input data characteristics, computational requirements, and quality metrics. Continuous stream of input data, varying network conditions, and computational requirements of different ML models create dynamic workload scenarios. These requirements include higher prediction accuracy of ML models, latency of inferencing results from ML models, network utilization, energy efficiency, resilience, and an overall higher quality of service. Further, multi-modal affective computing applications are prone to input data perturbations, which also presents an opportunity to exploit the inherent resilience to selectively process input data. This brings sensing-awareness into computation, and compute-awareness to sensing through bi-directional feedback. Cross-layered sense-compute co-optimization improves sensing, computation, and communication aspects of edge-ML based affective computing applications holistically.

8.1.1 Strength

The strength of this study may lie in four aspects: i) This is the first study that uses multi-modal signals from real postoperative adult patients for the purpose of developing an automatic pain assessment tool, as an exemplar case study for affective computing, for any single-modal model or a multi-modal model.

ii) The use of weak supervision in our data labeling process that had been previously unexplored in pain assessment studies or any affective computing studies. It eliminates the need for constantly asking patients for their pain levels and therefore reduces the burden placed on them during the trials.

iii) The use of feature selection in our procedure among with proper use of the machine learning methods helps determine the most informative features, reduces the complexity of our pain models, and able to account for inter-individual variability in pain responses. The evaluation results show significant improvements in comparison with the state-of-the-art (Werner et al [194] and Thiam et al [180]).

iv) The proposed closed-loop intelligent and adaptive multi-modal sensing framework for energy efficient and resilient affective computing applications guides sensing and sense-making by monitoring input signal quality and discrepancy detection via an intelligent control framework that reduces garbage data to achieve both energy efficiency as well as improved quality.

8.1.2 Limitations and alternative strategies

The limitations of this study may lie in four aspects:

i) The soft activities such as walking and lifting legs themselves in the presence of noise, in the form of motion artifacts and baseline wander, will cause change in physiological parameters such as HR increase in addition to pain. A control group without surgical, medical or any other type of pain may help understand or compare the influence of motion artifacts on the physiological parameters. Another alternative approach would be a unified activity flow design regardless of the surgical or pain area

ii) Imbalanced labels in each patient's data, since we did not collect data in a laboratory setting, most patients did not report all the different pain levels during the trials. Most noticeably, this led to a relatively smaller number of labeled examples from the highest pain level (PL4). This consequently decreased the performance accuracy for that pain category classification (BL vs PL4). In a controlled laboratory setting, one can design the study to force the pain intensity levels to be balanced, which is not feasible in real settings.

iii) Finding a significant difference between different pain levels in our study is difficult. We believe this is due to the fact that variations in ECG signals in response to different pain levels are much harder to distinguish in comparison to different pain levels versus baseline. Moreover, it is worth mentioning that the state of the art in pain assessment focuses on comparing baseline with other pain levels (eg, Werner et al [194]). We believe the reason is to find out if the patient has pain (baseline vs other pain levels).

iv) The relatively poor performances of the BL vs PL1 and BL vs PL4 models across both single and multimodal models are also understandable because they lie at the extremes of the pain threshold. The BL vs PL1 models might find it more challenging to distinguish between baseline levels and the lowest pain intensity due to the subtlety of the physiological responses collected while experiencing this pain level. The BL vs PL4, however, might find it challenging to distinguish pain levels due to the scarcity of such labels collected during trials. Data augmentation can help mitigate this problem, but there is no substitute for real data. Moreover, most single modality models for the highest pain intensity trained without any data augmentation techniques only performed as well as random guessing (˜50%) and sometimes even worse.

v) The sliding window technique for some modalities require a longer time interval to provide a better performance

vi) The perception of pain is induced by the interplay of the three motivational components:

sensory, affective, and cognitive. Some non-medicine pain relief techniques may affect the perceived pain level, in addition to pain medication. Hence, the sensory dimension of pain (pain intensity) and the affective dimension of pain (pain distress/unpleasantness) may be separately assessed in self-report using VAS or NRS as two labeling references.

vii) Even though having multiple labeling sources (self and objective reports) is an advantage in modeling, not all these reports provide labels at a frequency comparable to the signal or feature update rate. The solution would be using a compromised assessment frequency considering both the signal response time and clinical need. Machine learning techniques that rely on soft labels (such as probability or likelihood) or incomplete data labeling (i.e., unsupervised or semi-supervised learning) could be alternative solutions as well.

8.2 Significance of the study

The significance of this study to the affective computing field include:

- A review of the application level and system level perspectives of the affective computing services through the application of pain assessment and gives an overall picture of the field bridging the clinical and experimental points of views, a gap that has rarely been discussed previously;
- Design and development of an end-to-end platform for multimodal machine learning affective computing services and a thorough analysis of their roles in the prediction, performance and energy; this could provide a benchmark value for future studies;
- Summary on the importance of each part in the classic pattern recognition processing flow that contributes to pain estimation, which is usually discussed separately.

8.3 Future research directions

An application in real life may encounter different uncertainties. These uncertainties need to be evaluated and taken into consideration in future work. Pain assessment tools needs to be adapted and validated on a broader range of signals and different clinical populations for a better understanding. Studies across different databases or/and from the experimental to the clinical environment should be encouraged, although there still exist many challenges. In the following, we discuss promising research directions regarding (1) personalization, (2) data quality management, and (3) deep learning architectures.

Personalization

Since pain is a subjective experience that tends to have a large inter-individual variability, building a monolithic model for all patients might not be a viable solution. A promising future direction for this research study is to build personalized machine learning models that can benefit from using data from groups of similar patients, but which are fine-tuned to make predictions on a single person. Prior research has used multitask machine learning (MTL) to account for inter-individual variability and build personalized models for the task of mood prediction [178]. This is a feasible future research direction that would be applicable to the domain of pain assessment, not only for the acute pain of surgery but also for patients that experience chronic pain. We believe that personalized modeling will be a vital step in creating clinically viable pain assessment algorithms.

Data Quality Management

Input data quality is an essential component for improving prediction accuracy of ML-driven smart affective computing applications. Processing exclusively quality input data also improves the bandwidth utilization and latency of tasks run on the edge nodes. Some of the techniques presented in this thesis address the sensing aspects through continuous monitoring and analysis of input data quality. Qualitative assessment of sensory data can be improved significantly beyond the rule-based monitors using cognitive learning models. Design of autonomous models for input data quality management remains an open challenge. Autonomous models enable reasoning for different input perturbations to assess true quality of sensory inputs. Consequently, the garbage data that is unnecessarily processed is minimized, supporting the scalability of edge ML solutions. Quality assessment of input data is significant in other sensor-driven domains such as autonomous driving, robotics, and computer vision etc.

Deep Learning Architectures

Another future research direction would be to explore and build real-time multimodal pain assessment systems using deep learning architectures. In such scenarios, it is quite possible to have missing or incomplete data from one or more modalities. Moreover, real-time systems are limited by their computational complexity and power constraints. Therefore, with the help of the experiments performed in this study, we hope to build models that are able to dynamically determine which modalities to use in an energy-efficient manner without compromising on performance given the clinical context.

8.4 Applicability to Other Affective Computing Services

This study mainly was focused on pain as an exemplar. Positive and negative affective systems function either relatively independently, or inversely related (e.g. under conditions characterized by uncertainty, including pain and stress). The successful implementation of an affective computing application proves the general applicability of the solutions developed in this thesis. Furthermore, it shows that with the solutions provided in this work, the complete pipeline for physiology-based affective applications can be implemented, from sensing, feature extraction, classification and pain assessment to finally using the pain intensity information to adjust an affective application. With sensing devices implementing the developed sensor concept for acquiring physiological data using non-invasive technologies physiological changes can evolve in a matter of milliseconds, seconds, or even minutes. Some of these changes are temporary but some are permanent. Therefore, the time windows of change are of interest [189]. Moreover, affective signals are influenced by a variety of factors internally or externally. A literature review is needed to get know better the relation between physiological signals and the affective state. Finally, any developed method needs to be validated in terms of content, criteria-related, construct, and context [188].

Bibliography

[1] M. Abadi, , et al. Tensorflow: A system for large-scale machine learning. In *OSDI'16*, pages 265–283, 2016.

[2] S. Alam, S. Datta, A. D. Choudhury, and A. Pal. Sensor agnostic photoplethysmogram signal quality assessment using morphological analysis. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pages 176–185, 2017.

[3] J. Allen. Photoplethysmography and its application in clinical physiological measurement. *Physiol Meas.*, 28(3):1–39, 2007.

[4] N. S. Altman. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185, 1992.

[5] D. Amiri, A. Anzanpour, I. Azimi, M. Levorato, P. Liljeberg, N. Dutt, and A. M. Rahmani. Context-aware sensing via dynamic programming for edge-assisted wearable systems. *ACM Transactions on Computing for Healthcare*, 1(2):1–25, 2020.

[6] D. Amiri et al. Edge-Assisted Sensor Control in Healthcare IoT. In *IEEE GLOBECOM SAC EH*, pages 1–6, 2018.

[7] S. Anders, M. Lotze, M. Erb, W. Grodd, and N. Birbaumer. Brain activity underlying emotional valence and arousal: A response-related fmri study. *Human brain mapping*, 23(4):200–209, 2004.

[8] S. A. H. Aqajari, R. Cao, E. Kasaeyan Naeini, M.-D. Calderon, K. Zheng, N. Dutt, P. Liljeberg, S. Salanterä, A. M. Nelson, and A. M. Rahmani. Pain assessment tool with electrodermal activity for postoperative patients: Method validation study. *JMIR Mhealth Uhealth*, 9(5):e25258, May 2021.

[9] S. A. H. Aqajari, R. Cao, A. H. A. Zargari, and A. M. Rahmani. An end-to-end and accurate ppg-based respiratory rate estimation approach using cycle generative adversarial networks. *CoRR*, abs/2105.00594, 2021.

[10] S. A. H. Aqajari, E. K. Naeini, M. A. Mehrabadi, S. Labbaf, N. Dutt, and A. M. Rahmani. pyeda: An open-source python toolkit for pre-processing and feature extraction of electrodermal activity. *Procedia Computer Science*, 184:99–106, 2021.

[11] C. Arbour, M. Choinére, J. Topolovec-Vranic, C. G. Loiselle, and C. Célinas. Can fluctuations in vital signs be used for pain assessment in critically ill patients with a traumatic brain injury? In *Pain Research and Treatment*, volume 2014, page 175794, 2014.

[12] M. Arif-Rahu and M. J. Grap. Facial expression and pain in the critically ill non-communicative patient: state of science review. *Intensive and Critical Care Nursing*, 26(6):343–352, 2010.

[13] M. Asgari Mehrabadi, I. Azimi, F. Sarhaddi, A. Axelin, H. Niela-Vilén, S. Myllyntausta, S. Stenholm, N. Dutt, P. Liljeberg, and A. M. Rahmani. Sleep tracking of a commercially available smart ring and smartwatch against medical-grade actigraphy in everyday settings: Instrument validation study. *JMIR Mhealth Uhealth*, 8(10):e20465, Nov 2020.

[14] M. Aydede. Defending the iasp definition of pain. *The Monist*, 100(4):439–464, 2017.

[15] I. Azimi, O. Oti, S. Labbaf, H. Niela-Vilén, A. Axelin, N. Dutt, P. Liljeberg, and A. M. Rahmani. Personalized maternal sleep quality assessment: An objective iot-based longitudinal study. *IEEE Access*, 7:93433–93447, 2019.

[16] M. Baig and H. Gholamhosseini. Smart health monitoring systems: An overview of design and modeling. *Journal of Medical Systems*, 37(2):9898, 2013.

[17] J. Bakker, M. Pechenizkiy, and N. Sidorova. What's your current stress level? detection of stress patterns from gsr sensor data. In *2011 IEEE 11th international conference on data mining workshops*, pages 573–580. IEEE, 2011.

[18] J. Barr, G. L. Fraser, K. Puntillo, E. W. Ely, C. Gélinas, J. F. Dasta, J. E. Davidson, J. W. Devlin, J. P. Kress, A. M. Joffe, et al. Clinical practice guidelines for the management of pain, agitation, and delirium in adult patients in the intensive care unit. *Critical Care Medicine*, 41(1):263–306, 2013.

[19] M. Baruah and B. Banerjee. Modality selection for classification on time-series data. In *MileTS*, 2020.

[20] N. Ben-Israel, M. Kliger, G. Zuckerman, Y. Katz, and R. Edry. Monitoring the nociception level: a multi-parameter approach. *Journal of clinical monitoring and computing*, 27(6):659–668, 2013.

[21] M. Benedek et al. A continuous measure of phasic electrodermal activity. *Journal of neuroscience methods*, 2010.

[22] M. Benedek and C. Kaernbach. Decomposition of skin conductance data by means of nonnegative deconvolution. *psychophysiology*, 47(4):647–658, 2010.

[23] M. Bolanos, H. Nazeran, and E. Haltiwanger. Comparison of heart rate variability signal features derived from electrocardiography and photoplethysmography in healthy individuals. In *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4289–4294. IEEE, 2006.

[24] J. J. Bonica. The management of. *Pain (Philadelphia: Lea and Febiger, 1954)*, pages 1243–1244, 1953.

[25] W. Boucsein. *Electrodermal activity.* Springer Science & Business Media, 2012.

[26] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[27] H. Breivik, P. C. Borchgrevink, S. Allen, L. Rosseland, L. Romundstad, E. Hals, G. Kvarstein, and A. Stubhaug. Assessment of pain. In *British Journal of Anaesthesia*, volume 101, pages 17 – 24, 2008.

[28] H. Breivik, P.-C. Borchgrevink, S.-M. Allen, L.-A. Rosseland, L. Romundstad, E. Breivik Hals, G. Kvarstein, and A. Stubhaug. Assessment of pain. *BJA: British Journal of Anaesthesia*, 101(1):17–24, 2008.

[29] A. J. Camm, M. Malik, J. T. Bigger, G. Breithardt, S. Cerutti, R. Cohen, P. Coumel, E. Fallen, H. Kennedy, R. Kleiger, et al. Heart rate variability: standards of measurement, physiological interpretation and clinical use. task force of the european society of cardiology and the north american society of pacing and electrophysiology. *Circulation.*, 1996.

[30] R. Cao, S. A. H. Aqajari, E. K. Naeini, and A. M. Rahmani. Objective pain assessment using wrist-based ppg signals: A respiratory rate based method. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 1164–1167. IEEE, 2021.

[31] T. Cascino and M. J. Shea. Electrocardiography - cardiovascular disorders, Jul 2022.

[32] C. R. Chapman, S. Oka, D. H. Bradshaw, R. C. Jacobson, and G. W. Donaldson. Phasic pupil dilation response to noxious stimulation in normal volunteers: relationship to brain evoked potentials and pain report. *Psychophysiology*, 36(1):44–52, 1999.

[33] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[34] P. Cheang and P. Smith. An overview of non-contact photoplethysmography. *Electronic Systems and Control Division Research*, pages 57–9, 2003.

[35] A. Choi and H. Shin. Photoplethysmography sampling frequency: Pilot assessment of how low can we go to analyze pulse rate variability with reliability? *Physiological measurement*, 38(3):586, 2017.

[36] R. Chou, D. B. Gordon, O. A. de Leon-Casasola, J. M. Rosenberg, S. Bickler, T. Brennan, T. Carter, C. L. Cassidy, E. H. Chittenden, E. Degenhardt, et al. Management of postoperative pain: a clinical practice guideline from the american pain society, the american society of regional anesthesia and pain medicine, and the american society of anesthesiologists' committee on regional anesthesia, executive committee, and administrative council. *The journal of pain*, 17(2):131–157, 2016.

[37] I. I. Christov. Real time electrocardiogram qrs detection using combined adaptive threshold. *Biomedical engineering online*, 3(1):28, 2004.

[38] N. Constant et al. Pulse-glasses: An unobtrusive, wearable hr monitor with internet-of-things functionality. In *IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks*, pages 1–5, 2015.

[39] R. Cowen, M. K. Stasiowska, H. Laycock, and C. Bantel. Assessing pain objectively: the use of physiological markers. In *Anaesthesia*, volume 70, pages 828–847, 2015.

[40] T. R. Dawes, B. Eden-Green, C. Rosten, J. Giles, R. Governo, and F. Marcelline. Objectively measuring pain using facial expression: is the technology finally ready? In *Pain Management*, volume 8(2), pages 105–113, 2018.

[41] M. E. Dawson, A. M. Schell, and D. L. Filion. *The Electrodermal System*, page 217–243. Cambridge Handbooks in Psychology. Cambridge University Press, 4 edition, 2016.

[42] S. Devot, A. M. Bianchi, E. Naujoka, M. O. Mendez, A. Braurs, and S. Cerutti. Sleep monitoring through a textile recording system. In *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2560–2563. IEEE, 2007.

[43] L. Duch, S. Basu, R. Braojos, G. Ansaloni, L. Pozzi, and D. Atienza. Heal-wear: An ultra-low power heterogeneous system for bio-signal analysis. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 64(9):2448–2461, 2017.

[44] M. Elgendi. Optimal signal quality index for photoplethysmogram signals. *Bioengineering*, 3(4):21, 2016.

[45] Empatica. Medical devices, ai and algorithms for remote patient monitoring, 2015.

[46] W. A. H. Engelse and C. Zeelenberg. A single scan algorithm for qrs-detection and feature extraction. In *Proc IEEE Comp Cardiol*, pages 37–42, 1979.

[47] D. C. Fowles, M. J. Christie, R. Edelberg, W. W. Grings, D. T. Lykken, and P. H. Venables. Publication recommendations for electrodermal measurements. *Psychophysiology*, 18(3):232–239, 1981.

[48] A. J. Fridlund and J. T. Cacioppo. Guidelines for human electromyographic research. *Psychophysiology*, 23(5):567–589, 1986.

[49] N. Ganapathy, R. Swaminathan, and T. M. Deserno. Deep learning on 1-d biosignals: a taxonomy-based survey. *Yearbook of medical informatics*, 27(01):098–109, 2018.

[50] F. C. Geisler, N. Vennewald, T. Kubiak, and H. Weber. The impact of heart rate variability on subjective well-being is mediated by emotion regulation. *Personality and individual differences*, 49(7):723–728, 2010.

[51] C. Gélinas and C. Johnston. Pain assessment in the critically ill ventilated adult: validation of the critical-care pain observation tool and physiologic indicators. *The Clinical journal of pain*, 23(6):497–505, 2007.

[52] C. Gélinas, K. A. Puntillo, P. Levin, and E. Azoulay. The behavior pain assessment tool for critically ill adults: a validation study in 28 countries. *Pain*, 158(5):811–821, 2017.

[53] S. Geuter, M. Gamer, S. Onat, and C. Büchel. Parametric trial-by-trial prediction of pain by easily available physiological measures. In *Pain*, volume 155, pages 994–1001, 2014.

[54] C.-H. Goh, L. K. Tan, N. H. Lovell, S.-C. Ng, M. P. Tan, and E. Lim. Robust ppg motion artifact detection using a 1-d convolution neural network. *Computer methods and programs in biomedicine*, 196:105596, 2020.

[55] P. Gomes, P. Margaritoff, and H. Silva. pyHRV: Development and evaluation of an open-source python toolbox for heart rate variability (HRV). In *Proc. Int'l Conf. on Electrical, Electronic and Computing Engineering (IcETRAN)*, pages 822–828, 2019.

[56] A. Greco, G. Valenza, A. Lanata, E. P. Scilingo, and L. Citi. cvxeda: A convex optimization approach to electrodermal activity processing. *IEEE Transactions on Biomedical Engineering*, 63(4):797–804, 2015.

[57] J. Gregory and L. McGowan. An examination of the prevalence of acute pain for hospitalised adult patients: a systematic review. *Journal of Clinical Nursing*, 25(5-6):583–598, 2016.

[58] S. Gruss, R. Treister, P. Werner, and H. C. Traue. Pain intensity recognition rates via biopotential feature patterns with support vector machines. In *PLOS ONE*, volume 10(10), 2015.

[59] J. Gubbi et al. Internet of things (iot): A vision, architectural elements, and future directions. *Future Gener Comput Syst*, 29(7):1645–60, 2013.

[60] H. Gunes and M. Piccardi. Affect recognition from face and body: early fusion vs. late fusion. In *2005 IEEE international conference on systems, man and cybernetics*, volume 4, pages 3437–3443. IEEE, 2005.

[61] I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh. *Feature extraction: foundations and applications*, volume 207. Springer, 2008.

[62] P. Hamilton. Open source ecg analysis. In *Computers in cardiology*, pages 101–104. IEEE, 2002.

[63] K. Hamunen, V. Kontinen, E. Hakala, P. Talke, M. Paloheimo, and E. Kalso. Effect of pain on autonomic nervous system indices derived from photoplethysmography in healthy volunteers. In *British Journal of Anaesthesia*, volume 108, pages 838–844, 2012.

[64] H. Han et al. Development of real-time motion artifact reduction algorithm for a wearable photoplethysmography. In *Conf Proc IEEE Eng Med Biol Soc.*, pages 1538–1541, 2007.

[65] A. J. Hautala, J. Karppinen, and T. Seppänen. Short-term assessment of autonomic nervous system as a potential tool to quantify pain experience. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2684–2687. IEEE, 2016.

[66] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[67] K. Heikkinen, S. Salanterä, M. Kettu, and M. Taittonen. Prostatectomy patients' postoperative pain assessment in the recovery room. *Journal of advanced nursing*, 52(6):592–600, 2005.

[68] K. Herr, P. J. Coyne, M. McCaffery, R. Manworren, and S. Merkel. Pain assessment in the patient unable to self-report: position statement with clinical practice recommendations. *Pain management nursing*, 12(4):230–250, 2011.

[69] E. Ilana. Pain terms; a list with definitions and notes on usage. *Pain*, 6:249, 1979.

[70] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[71] M. Jiang, R. Mieronkoski, A. M. Rahmani, N. Hagelberg, S. Salanterä, and P. Liljeberg. Ultra-short-term analysis of heart rate variability for real-time acute pain monitoring with wearable electronics. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1025–1032. IEEE, 2017.

[72] A. John, B. Cardiff, and D. John. A generalized signal quality estimation method for iot sensors. *arXiv preprint arXiv:2002.01279*, 2020.

[73] A. M. Kabes, J. K. Graves, and J. Norris. Further validation of the nonverbal pain scale in intensive care patients. *Critical care nurse*, 29(1):59–66, 2009.

[74] M. Kächele et al. Multimodal data fusion for person-independent, continuous estimation of pain intensity. In *EANN*, 2015.

[75] M. Kächele, P. Thiam, M. Amirian, P. Werner, S. Walter, F. Schwenker, and G. Palm. Multimodal data fusion for person-independent, continuous estimation of pain intensity. In *International Conference on Engineering Applications of Neural Networks*, pages 275–285. Springer, 2015.

[76] S. Kaltwang, O. Rudovic, and M. Pantic. Continuous pain intensity estimation from facial expressions. In *International Symposium on Visual Computing*, pages 368–377. Springer, 2012.

[77] E. Kasaeyan Naeini et al. An edge-assisted and smart system for real-time pain monitoring. In *CHASE*, 2019.

[78] E. Kasaeyan Naeini, M. Jiang, E. Syrjälä, M.-D. Calderon, R. Mieronkoski, K. Zheng, N. Dutt, P. Liljeberg, S. Salanterä, A. M. Nelson, and A. M. Rahmani. Prospective study evaluating a pain assessment tool in a postoperative environment: Protocol for algorithm testing and enhancement. *JMIR Res Protoc*, 9(7):e17783, Jul 2020.

[79] E. Kasaeyan Naeini, A. Subramanian, M.-D. Calderon, K. Zheng, N. Dutt, P. Liljeberg, S. Salantera, A. M. Nelson, and A. M. Rahmani. Pain recognition with electrocardiographic features in postoperative patients: Method validation study. *J Med Internet Res*, 23(5):e25079, May 2021.

[80] Y. Kato, C. J. Kowalski, and C. S. Stohler. Habituation of the early pain-specific respiratory response in sustained pain. *Pain*, 91(1-2):57–63, 2001.

[81] M. L. Kent, P. J. Tighe, I. Belfer, T. J. Brennan, S. Bruehl, C. M. Brummett, C. C. Buckenmaier, A. Buvanendran, R. I. Cohen, P. Desjardins, et al. The acttion–aps–aapm pain taxonomy (aaapt) multidimensional approach to classifying acute pain conditions. *Pain Medicine*, 18(5):0–0, 2017.

[82] E. J. Keogh and M. J. Pazzani. Scaling up dynamic time warping for datamining applications. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 285–289, 2000.

[83] L. Kessous, G. Castellano, and G. Caridakis. Multimodal emotion recognition in speech-based interaction using facial expression, body gesture and acoustic analysis. *Journal on Multimodal User Interfaces*, 3(1):33–48, 2010.

[84] M. A. Khan and N. Alkaabi. Rebirth of distributed ai—a review of ehealth research. *Sensors*, 21(15):4999, 2021.

[85] D. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980, 2014.

[86] D. G. Klein, M. Dumpe, E. Katz, and J. Bena. Pain assessment in the intensive care unit: development and psychometric testing of the nonverbal pain assessment tool. *Heart & Lung*, 39(6):521–528, 2010.

[87] R. Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.

[88] I. Korhonen and A. Yli-Hankala. Photoplethysmography and nociception. *Acta Anaesthesiologica Scandinavica*, 53(8):975–985, 2009.

[89] M. Kunz, K. Miriam, M. Veit, S. Karsten, and L. Stefan. On the relationship between self-report and facial expression of pain. In *Journal of Pain*, volume 5, pages 368–376, 2004.

[90] J. Laitala, M. Jiang, E. Syrjälä, E. K. Naeini, A. Airola, A. M. Rahmani, N. D. Dutt, and P. Liljeberg. Robust ecg r-peak detection using lstm. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pages 1104–1111, 2020.

[91] Q. V. Le et al. A tutorial on deep learning part 2: Autoencoders, convolutional neural networks and recurrent neural networks. *Google Brain*, 20:1–20, 2015.

[92] Y. LeCun and Y. Bengio. *The Handbook of Brain Theory and Neural Networks*, chapter Convolutional Networks for Images, Speech, and Time Series, pages 255–258. MIT Press, 1998.

[93] J. S. Lee, E. Hobden, I. G. Stiell, and G. A. Wells. Clinically important change in the visual analog scale after adequate pain control. *Academic Emergency Medicine*, 10(10):1128–1130, 2003.

[94] G. Lemaître, F. Nogueira, and C. K. Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18(17):1–5, 2017.

[95] K. Li et al. Onboard tagging for real-time quality assessment of photoplethysmograms acquired by a wireless reflectance pulse oximeter. *IEEE Transactions on Biomedical Circuits and Systems*, 6(1):54–63, 2012.

[96] Q. Li and G. D. Clifford. Dynamic time warping and machine learning for signal quality assessment of pulsatile signals. *Physiol. Meas.*, 33:1491–1501, 2012.

[97] S. Liu et al. On-demand deep model compression for mobile devices: A usage-driven model selection framework. In *MobiSys*, 2018.

[98] D. Lopez-Martinez, K. Peng, S. C. Steele, A. J. Lee, D. Borsook, and R. Picard. Multi-task multiple kernel machines for personalized pain recognition from functional near-infrared spectroscopy brain signals. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 2320–2325. IEEE, 2018.

[99] A. Lourenço, H. Silva, P. Leite, R. Lourenço, and A. L. Fred. Real time electrocardiogram segmentation for finger based ecg biometrics. In *Biosignals*, pages 49–54, 2012.

[100] P. Lucey, L. Patrick, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and C. Sien. Painful monitoring: automatic pain monitoring using the unbc-mcmaster shoulder pain expression archive database. In *Image and Vision Computing*, volume 30, pages 197–205, 2012.

[101] G. M. et al. A comprehensive survey on multimodal medical signals fusion for smart healthcare systems. *Information Fusion*, 2021.

[102] M. Ma, J. Ren, L. Zhao, S. Tulyakov, C. Wu, and X. Peng. Smil: Multimodal learning with severely missing modality. *arXiv preprint arXiv:2103.05677*, 2021.

[103] K. V. Madhav, M. R. Ram, E. H. Krishna, N. R. Komalla, and K. A. Reddy. Estimation of respiration rate from ecg, bp and ppg signals using empirical mode decomposition. In *2011 IEEE International Instrumentation and Measurement Technology Conference*, pages 1–4. IEEE, 2011.

[104] A. Mahmoudzadeh, I. Azimi, A. M. Rahmani, and P. Liljeberg. Lightweight photoplethysmography quality assessment for real-time iot-based health monitoring using unsupervised anomaly detection. *Procedia Computer Science*, 184:140–147, 2021.

[105] S. Majumder, T. Mondal, and M. J. Deen. Wearable sensors for remote health monitoring. *Sensors*, 17(1):130, 2017.

[106] Masinelli et al. Self-aware machine learning for multimodal workload monitoring during manual labor on edge wearable sensors. *Design&Test*, 2020.

[107] R. McCraty and F. Shaffer. Heart rate variability: new perspectives on physiological mechanisms, assessment of self-regulatory capacity, and health risk. *Global advances in health and medicine*, 4(1):46–61, 2015.

[108] R. Mieronkoski, I. Azimi, A. Rahmani, R. Aantaa, V. Terävä, P. Liljeberg, and S. Salanterä. The internet of things for basic nursing care—a scoping review. *International Journal of Nursing Studies*, 69:78 – 90, 2017.

[109] R. Mieronkoski et al. The internet of things for basic nursing care—a scoping review. *International Journal of Nursing Studies*, 69:78–90, 2017.

[110] P. Myles, D. Myles, W. Galagher, D. Boyd, C. Chew, N. MacDonald, and A. Dennis. Measuring acute postoperative pain using the visual analog scale: the minimal clinically important difference and patient acceptable symptom state. *BJA: British Journal of Anaesthesia*, 118(3):424–429, 2017.

[111] E. K. Naeini, I. Azimi, A. M. Rahmani, P. Liljeberg, and N. Dutt. A real-time ppg quality assessment approach for healthcare internet-of-things. *Procedia Computer Science*, 151(C), 2019.

[112] E. K. Naeini, S. Shahhosseini, A. Kanduri, P. Liljeberg, A. M. Rahmani, and N. Dutt. Amser: Adaptive multimodal sensing for energy efficient and resilient ehealth systems. In *2022 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 1455–1460. IEEE, 2022.

[113] J. Ngiam et al. Multimodal deep learning. In *ICML*, 2011.

[114] C. O'Keeffe et al. Role of ambulance response times in the survival of patients with out-of-hospital cardiac arrest. *EMJ*, 2011.

[115] C. Orphanidou, T. Bonnici, P. Charlton, D. Clifton, D. Vallance, and L. Tarassenko. Signal-quality indices for the electrocardiogram and photoplethysmogram: Derivation and applications to wireless monitoring. *IEEE journal of biomedical and health informatics*, 19(3):832–838, 2014.

[116] C. Orphanidou et al. Signal-quality indices for the ecg and ppg: Derivation and applications to wireless monitoring. *JBHI*, 2015.

[117] S. M. Oteafy. A framework for heterogeneous sensing in big sensed data. In *GLOBECOM*, 2016.

[118] J. Pan and W. J. Tompkins. A real-time qrs detection algorithm. *IEEE transactions on biomedical engineering*, 32(3):230–236, 1985.

[119] A. Pantelopoulos and N. G. Bourbakis. A survey on wearable sensor-based systems for health monitoring and prognosis. *Trans. Sys. Man Cyber Part C*, 40(1):1–12, 2010.

[120] G. B. Papini, P. Fonseca, X. L. Aubert, S. Overeem, J. W. Bergmans, and R. Vullings. Photoplethysmography beat detection and pulse morphology quality assessment for signal reliability estimation. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 117–120. IEEE, 2017.

[121] J. Park, S. Samarakoon, M. Bennis, and M. Debbah. Wireless network intelligence at the edge. *Proceedings of the IEEE*, 107(11):2204–2239, 2019.

[122] J. F. Payen, J. L. Bosson, G. Chanques, J. Mantz, and J. Labarere. Pain assessment is associated with decreased duration of mechanical ventilation in the intensive care unita post hocanalysis of the dolorea study. *Anesthesiology: The Journal of the American Society of Anesthesiologists*, 111(6):1308–1316, 2009.

[123] J.-F. Payen, O. Bru, J.-L. Bosson, A. Lagrasta, E. Novel, I. Deschaux, P. Lavagne, and C. Jacquot. Assessing pain in critically ill sedated patients by using a behavioral pain scale. *Critical care medicine*, 29(12):2258–2263, 2001.

[124] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[125] T. Pereira, P. R. Almeida, J. P. Cunha, and A. Aguiar. Heart rate variability metrics for fine-grained stress level assessment. *Computer methods and programs in biomedicine*, 148:71–80, 2017.

[126] T. Pereira, C. Ding, K. Gadhoumi, N. Tran, R. A. Colorado, K. Meisel, and X. Hu. Deep learning approaches for plethysmography signal quality assessment in the presence of atrial fibrillation. *Physiological measurement*, 40(12):125002, 2019.

[127] T. Pereira, K. Gadhoumi, M. Ma, R. Colorado, K. J. Keenan, K. Meisel, and X. Hu. Robust assessment of photoplethysmogram signal quality in the presence of atrial fibrillation. In *2018 Computing in Cardiology Conference (CinC)*, volume 45, pages 1–4. IEEE, 2018.

[128] T. Pereira, K. Gadhoumi, M. Ma, L. Xiuyun, R. Xiao, R. A. Colorado, K. J. Keenan, K. Meisel, and X. Hu. A supervised approach to robust photoplethysmography quality assessment. *IEEE Journal of Biomedical and Health Informatics*, 2019.

[129] C. Perera et al. Context aware computing for the internet of things: A survey. *IEEE communications surveys & tutorials*, 2013.

[130] M. T. Petterson, V. L. Begnoche, and J. M. Graybeal. The effect of motion on pulse oximetry and its clinical significance. *Anesthesia* & *Analgesia*, 105(6):S78–S84, 2007.

[131] J. Piskorski and P. Guzik. Filtering poincare plots. *Computational methods in science and technology*, 11(1):39–48, 2005.

[132] H. F. Posada-Quintero et al. Electrodermal activity is sensitive to cognitive stress under water. *Frontiers in Physiology*, 2018.

[133] B. Pourghebleh et al. Data aggregation mechanisms in the internet of things: A systematic review of the literature and recommendations for future research. *Journal of Network and Computer Applications*, 2017.

[134] K. M. Prkachin. The consistency of facial expressions of pain: a comparison across modalities. *Pain*, 51(3):297–306, 1992.

[135] K. M. Prkachin. Assessing pain by facial expression: facial expression as nexus. *Pain Research and Management*, 14(1):53–58, 2009.

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

[137] K. A. Puntillo, A. B. Morris, C. L. Thompson, J. Stanik-Hutt, C. A. White, and L. R. Wild. Pain behaviors observed during six common procedures: results from thunder project ii. *Critical care medicine*, 32(2):421–427, 2004.

[138] V. P. Rachim and W.-Y. Chung. Wearable noncontact armband for mobile ecg monitoring system. *IEEE transactions on biomedical circuits and systems*, 10(6):1112–1118, 2016.

[139] W. Raffaeli and E. Arnaudo. Pain as a disease: an overview. *Journal of pain research*, 10:2003, 2017.

[140] M. A. Rahu, M. J. Grap, J. F. Cohn, C. L. Munro, D. E. Lyon, and C. N. Sessler. Facial expression as an indicator of pain in critically ill intubated adults during endotracheal suctioning. *American Journal of Critical Care*, 22(5):412–422, 2013.

[141] Ramshur. Ecg viewer, Retrieved on March 2020. `https://github.com/jramshur/ECG_Viewer/`.

[142] M. Rani et al. A systematic review of compressive sensing: Concepts, implementations and applications. *IEEE Access*, 2018.

[143] A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Ré. Snorkel: Rapid training data creation with weak supervision. In *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, volume 11, page 269. NIH Public Access, 2017.

[144] G. N. K. Reddy, M. S. Manikandan, and N. N. Murty. On-device integrated ppg quality assessment and sensor disconnection/saturation detection system for iot health monitoring. *IEEE Transactions on Instrumentation and Measurement*, 2020.

[145] D. Roh and H. Shin. Recurrence plot and machine learning for signal quality assessment of photoplethysmogram in mobile environment. *Sensors*, 21(6):2188, 2021.

[146] A. Ross. *Fusion, Feature-Level*, pages 597–602. Springer US, Boston, MA, 2009.

[147] M. Rossi, A. Cividjian, M. Fevre, M. Oddoux, J. Carcey, C. Halle, M. Frost, M. Gardellin, J. Payen, and L. Quintin. A beat-by-beat, on-line, cardiovascular index, cardean, to assess circulatory responses to surgery: a randomized clinical trial during spine surgery. *Journal of clinical monitoring and computing*, 26(6):441–449, 2012.

[148] M. S. Roy, R. Gupta, and K. D. Sharma. Photoplethysmogram signal quality evaluation by unsupervised learning approach. In *2020 IEEE Applied Signal Processing Conference (ASPCON)*, pages 6–10. IEEE, 2020.

[149] E. Sabeti, N. Reamaroon, M. Mathis, J. Gryak, M. Sjoding, and K. Najarian. Signal quality measure for pulsatile physiological signals using morphological features: Applications in reliability measure for pulse oximetry. *Informatics in Medicine Unlocked*, 16:100222, 2019.

[150] L. Salahuddin, J. Cho, M. G. Jeong, and D. Kim. Ultra short term analysis of heart rate variability for monitoring mental stress in mobile settings. In *2007 29th annual international conference of the ieee engineering in medicine and biology society*, pages 4656–4659. IEEE, 2007.

[151] Samsung. Samsung gear sport smartwatch, Retrieved on March 2020. `https://www.samsung.com/global/galaxy/gear-sport/`.

[152] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[153] V. K. Sarker, M. Jiang, T. N. Gia, A. Anzanpour, A. M. Rahmani, and P. Liljeberg. Portable multipurpose bio-signal acquisition and wireless streaming device for wearables. In *2017 IEEE sensors applications symposium (SAS)*, pages 1–6. IEEE, 2017.

[154] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[155] P. Schmidt et al. Introducing wesad, a multimodal dataset for wearable stress and affect detection. In *ICMI*, 2018.

[156] I. Selesnick and C. Burrus. Generalized digital butterworth filter design. *IEEE Transactions on signal processing*, 46(6):1688–1694, 1998.

[157] N. Selvaraj, A. Jaryal, J. Santhosh, K. K. Deepak, and S. Anand. Assessment of heart rate variability derived from finger-tip photoplethysmography as compared to electrocardiography. *Journal of medical engineering* & *technology*, 32(6):479–484, 2008.

[158] T. Sen and H. Shen. Machine learning based timeliness-guaranteed and energy-efficient task assignment in edge computing systems. In *2019 IEEE 3rd International Conference on Fog and Edge Computing (ICFEC)*, pages 1–10. IEEE, 2019.

[159] H. S. Seok, S. Han, J. Park, D. Roh, and H. Shin. Photoplethysmographic pulse quality assessment methods based on similarity analysis. In *2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS)*, pages 350–353. IEEE, 2018.

[160] M. Sesay, G. Robin, P. Tauzin-Fin, O. Sacko, E. Gimbert, J.-R. Vignes, D. Liguoro, and K. Nouette-Gaulain. Responses of heart rate variability to acute pain after minor spinal surgery: optimal thresholds and correlation with the numeric rating scale. *Journal of neurosurgical anesthesiology*, 27(2):148–154, 2015.

[161] F. Shaffer and J. P. Ginsberg. An overview of heart rate variability metrics and norms. *Frontiers in public health*, page 258, 2017.

[162] S. Shahhosseini, D. Seo, A. Kanduri, T. Hu, S.-S. Lim, B. Donyanavard, A. M. Rahmani, and N. Dutt. Online learning for orchestration of inference in multi-user end-edge-cloud networks. *ACM Transactions on Embedded Computing Systems (TECS)*, 2022.

[163] Y. Shi, G. Ding, H. Wang, H. E. Roman, and S. Lu. The fog computing service for healthcare. In *2015 2nd International Symposium on Future Information and Communication Technologies for Ubiquitous HealthCare (Ubi-HealthTech)*, pages 1–5. IEEE, 2015.

[164] Shimmer. Ecg sensor development kit, wearable ecg sensor, wireless ecg, Retrieved on January 2019. `https://www.shimmersensing.com/products/ecg-development-kit`.

[165] K. Sikka, A. A. Ahmed, D. Diaz, M. S. Goodwin, K. D. Craig, and M. S. Bartlett. Automated assessment of children's postoperative pain using computer vision. In *Pediatrics*, volume 136, pages e124–e131, 2015.

[166] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[167] M. Sokolova and G. Lapalme. A systematic analysis of performance measures for classification tasks. *Information processing & management*, 45(4):427–437, 2009.

[168] M. J. Solana, J. Lopez-Herce, S. Fernandez, R. Gonzalez, J. Urbano, and J. Lopez. Assessment of pain in critically ill children. is cuta-neous conductance a reliable tool? In *Journal of Critical Care*, volume 30, pages 481–485, 2015.

[169] J. Song, D. Li, X. Ma, G. Teng, and J. Wei. Pqr signal quality indexes: A method for real-time photoplethysmogram signal quality estimation based on noise interferences. *Biomedical Signal Processing and Control*, 47:88–95, 2019.

[170] J. T. Soto and E. Ashley. Deepbeat: A multi-task deep learning approach to assess signal quality and arrhythmia detection in wearable devices. *arXiv preprint arXiv:2001.00155*, 2020.

[171] G. Stewart and A. Panickar. Role of the sympathetic nervous system in pain. *Anaesthesia & Intensive Care Medicine*, 14(12):524–527, 2013.

[172] M. Stites. Observational pain scales in critically ill adults. *Critical Care Nurse*, 33(3):68–78, 2013.

[173] M. Stites. Observational pain scales in critically ill adults. *Critical care nurse*, 33(3):68–78, 2013.

[174] X. Sun et al. Assessment of photoplethysmogram signal quality using morphology integrated with temporal information approach. In *Conf. of the IEEE EMBS*, pages 3456–3459, 2012.

[175] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction.* MIT press, 2018.

[176] T. Tamura et al. Wearable photoplethysmographic sensors—past and present. *Electronics*, 3(2):282–302, 2014.

[177] J. Tang, S. Alelyani, and H. Liu. Feature selection for classification: A review. *Data classification: Algorithms and applications*, page 37, 2014.

[178] S. Taylor, N. Jaques, E. Nosakhare, A. Sano, and R. Picard. Personalized multitask learning for predicting tomorrow's mood, stress, and health. *IEEE Transactions on Affective Computing*, 11(2):200–213, 2017.

[179] X. Teng and Y. Zhang. Study on the peak interval variability of photoplethysmogtaphic signals. In *IEEE EMBS Asian-Pacific Conference on Biomedical Engineering, 2003.*, pages 140–141. IEEE, 2003.

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

[181] I. S. Thong, M. P. Jensen, J. Miró, and G. Tan. The validity of pain intensity measures: what do the nrs, vas, vrs, and fps-r measure? *Scandinavian journal of pain*, 18(1):99–107, 2018.

[182] P. J. Tighe, C. A. Harle, R. W. Hurley, H. Aytug, A. P. Boezaart, and R. B. Fillingim. Teaching a machine to feel postoperative pain: combining high-dimensional clinical data with machine learning algorithms to forecast acute postoperative pain. *Pain Medicine*, 16(7):1386–1401, 2015.

[183] D. A. Tompkins, J. G. Hobelmann, and P. Compton. Providing chronic pain management in the "fifth vital sign" era: Historical and treatment perspectives on a modern-day medical dilemma. In *Drug and Alcohol Dependence*, volume 173, pages S11 – S21, 2017.

[184] R. Treede. Transduction and transmission properties of primary nociceptive afferents. *Rossiiskii fiziologicheskii zhurnal imeni IM Sechenova*, 85(1):205–211, 1999.

[185] R. Treede, W. Rief, A. Barke, Q. Aziz, M. Bennett, R. Benoliel, M. Cohen, S. Evers, N. Finnerup, M. First, et al. Lavand'homme. *P., Nicholas, M., Perrot, S., Scholz, J., Shug, S., Smith, BH, Svensson, P., Vlaeyen, WS, and Wang SJ*, pages 1003–1007, 2015.

[186] K. Tyapochkin, E. Smorodnikova, and P. Pravdin. Smartphone ppg: signal processing, quality assessment, and impact on hrv parameters. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 4237–4240. IEEE, 2019.

[187] S. Vadrevu and M. S. Manikandan. A new quality-aware quality-control data compression framework for power reduction in iot and smartphone ppg monitoring devices. *IEEE Sensors Letters*, 3(7):1–4, 2019.

[188] E. L. van den Broek, J. H. Janssen, and J. H. Westerink. Guidelines for affective signal processing (asp): from lab to life. In *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pages 1–6. IEEE, 2009.

[189] E. L. van den Broek and J. H. Westerink. Considerations for emotion-aware consumer products. *Applied ergonomics*, 40(6):1055–1064, 2009.

[190] P. van Gent, H. Farah, N. van Nes, and B. van Arem. Heartpy: A novel heart rate algorithm for the analysis of noisy signals. *Transportation research part F: traffic psychology and behaviour*, 66:368–378, 2019.

[191] T. Voepel-Lewis, J. Zanotti, J. A. Dammeyer, and S. Merkel. Reliability and validity of the face, legs, activity, cry, consolability behavioral tool in assessing acute pain in critically ill patients. *American journal of critical care*, 19(1):55–61, 2010.

[192] Z. Wang and T. Oates. Imaging time-series to improve classification and imputation. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, pages 3939–3945, 2015.

[193] N. Wells, C. Pasero, and M. McCaffery. Improving the quality of care through pain assessment and management. *Patient safety and quality: An evidence-based handbook for nurses*, 2008.

[194] P. Werner, A. Al-Hamadi, R. Niese, S. Walter, S. Gruss, and H. C. Traue. Automatic pain recognition from video and biomedical signals. In *2014 22nd International Conference on Pattern Recognition*, pages 4582–4587. IEEE, 2014.

[195] P. Werner et al. Automatic recognition methods supporting pain assessment: A survey. *IEEE Transactions on Affective Computing*, 2019.

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

[197] S. S. Xu, M.-W. Mak, and C.-C. Cheung. Towards end-to-end ecg classification with raw signal extraction and deep neural networks. *IEEE journal of biomedical and health informatics*, 23(4):1574–1584, 2018.

[198] G. Yang et al. A health-iot platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box. *IEEE Transactions on Industrial Informatics*, 2014.

[199] G. Zamzmi, R. Kasturi, D. Goldgof, R. Zhi, T. Ashmeade, and Y. Sun. A review of automated pain assessment in infants: features, classification tasks, and databases. *IEEE reviews in biomedical engineering*, 11:77–96, 2017.

[200] L. Zhang. *Analysis of machine learning algorithms for the recognition of basic emotions: data mining of psychophysiological sensor information*. PhD thesis, Universität Ulm, 2019.

[201] Y. Zhang and J. Pan. Assessment of photoplethysmogram signal quality based on frequency domain and time series parameters. In *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5. IEEE, 2017.

[202] Z. Zhang. Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, pages 1–2. IEEE, 2018.

[203] C. Zong and R. Jafari. Robust heart rate estimation using wrist-based ppg signals in the presence of intense physical activities. In *IEEE EMBC*, pages 8078–8082, 2015.

[204] S. M. Zwakhalen, J. P. Hamers, H. H. Abu-Saad, and M. P. Berger. Pain in elderly people with severe dementia: a systematic review of behavioural pain assessment tools. *BMC geriatrics*, 6(1):1–15, 2006.