Assessing Cyclist's Stress on A Large-Scale: A Practical Smartphone-Based Data-Driven Approach

Ming Li, PhD
Shuchi Deb, PhD
Chris Le Dantec, PhD
Chaowei Wang
Aairish Singh

FINAL REPORT

CEDD
Center for Transportation Equity, Decisions & Dollars

ASSESSING CYCLIST'S STRESS ON A LARGE-SCALE: A PRACTICAL SMARTPHONE-BASED DATA-DRIVEN APPROACH

FINAL PROJECT REPORT

By:

Ming Li, PhD
Shuchi Deb, PhD
Chaowei Wang
Aairish Singh
University of Texas at Arlington

Chris Le Dantec, PhD
Georgia Institute of Technology

Sponsorship:
CTEDD

For:
Center for Transportation, Equity, Decisions and Dollars **(CTEDD)**
USDOT University Transportation Center
The University of Texas at Arlington
Woolf Hall, Suite 325

Acknowledgments

This work was supported by a grant from the Center for Transportation Equity, Decisions, and Dollars (CTEDD) funded by U.S. Department of Transportation Research and Innovative Technology Administration (OST-R) and housed at The University of Texas at Arlington.

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The Center for Transportation, Equity, Decisions and Dollars (CTEDD), the U.S. Government and matching sponsor assume no liability for the contents or use thereof.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
CTEDD 022-023		

4. Title and Subtitle	5. Report Date
Assessing Cyclist's Stress on A Large-Scale: A Practical Smartphone-Based Data-Driven Approach	6/23/2024
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Ming Li, PhD; ORCID: 0000-0002-4303-2438 Shuchi Deb, PhD; ORCID: 0000-0001-5626-2825 Chris Le Dantec; ORCID: 0000-0001-8338-4852	

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
University of Texas at Arlington The Georgia Institute of Technology	**11. Contract or Grant No.** USDOT - 69A3551747134

12. Sponsoring Organization Name and Address	13. Type of Report and Period Covered
Center for Transportation, Equity, Decisions and Dollars (CTEDD) USDOT University Transportation Center The University of Texas at Arlington Woolf Hall, Suite 325 Arlington TX 76019 United States	Final Report **14. Sponsoring Agency Code**

15. Supplementary Notes

Additional project information is available at Research Projects - Center for Transportation Equity, Decisions & Dollars (uta.edu)
Project performed under a grant from the U.S. Department of Transportation's University Transportation Centers (UTC) Program.

16. Abstract

Cycling presents a compelling solution for promoting personal health and environmental well-being, particularly for short-distance travel. Despite its numerous advantages, cycling uptake in the United States remains disproportionately low, primarily due to safety concerns. Traditional frameworks for assessing cyclist stress are hindered by their impracticality and inability to provide real-time evaluations. Self-report surveys and physiological measurements offer alternative approaches but suffer from limitations such as retrospective reporting biases and accessibility challenges, respectively. This project introduces CyclistAI, a novel smartphone-based cyclist stress assessment model that leverages context sensing. By combining Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) techniques, CyclistAI aims to provide real-time stress assessment for cyclists. Challenges in dataset annotation due to safety concerns are addressed through simulation-based data generation. Subsequent Domain Adaptation using Contrastive Learning bridges the simulation-to-reality gap and ensures the model's efficacy in real-world scenarios. In-field testing of CyclistAI demonstrates promising results, showcasing its potential as a pioneering stress assessment tool. Furthermore, this project proposes utilizing aggregated assessment results to create stress distribution maps, facilitating informed decision-making for urban planners to enhance cycling infrastructure and promote safer, more sustainable transportation environments.

17. Key Words	18. Distribution Statement
Cyclist stress assessment, cyclist safety, stress map, AI	No restrictions. This document is available to the public through the Transport Research International Documentation (TRID) repository.

19. Security Classification (of this report)	20. Security Classification (of this page)	21. No. of Pages	22. Price
Unclassified.	Unclassified.	56	N/A

Abstract

Cycling presents a compelling solution for promoting personal health and environmental well-being, particularly for short-distance travel. Despite its numerous advantages, cycling uptake in the United States remains disproportionately low, primarily due to safety concerns. Traditional frameworks for assessing cyclist stress are hindered by their impracticality and inability to provide real-time evaluations. Self-report surveys and physiological measurements offer alternative approaches but suffer from limitations such as retrospective reporting biases and accessibility challenges, respectively. This project introduces CyclistAI, a novel smartphone-based cyclist stress assessment model that leverages context sensing. By combining Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) techniques, CyclistAI aims to provide real-time stress assessment for cyclists. Challenges in dataset annotation due to safety concerns are addressed through simulation-based data generation. Subsequent Domain Adaptation using Contrastive Learning bridges the simulation-to-reality gap and ensures the model's efficacy in real-world scenarios. In-field testing of CyclistAI demonstrates promising results, showcasing its potential as a pioneering stress assessment tool. Furthermore, this project proposes utilizing aggregated assessment results to create stress distribution maps, facilitating informed decision-making for urban planners to enhance cycling infrastructure and promote safer, more sustainable transportation environments.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

Chapter 1: INTRODUCTION

Cycling represents a beneficial mode of transportation, fostering both personal health and environmental well-being, especially for short distances. By opting for two wheels over four, individuals contribute to reduced vehicle emissions, improved air quality, and decreased traffic congestion. Additionally, cycling promotes physical activity, enhancing cardiovascular health and mental well-being. Despite these advantages, its uptake remains disproportionately low in the United States. Only 1% of trips are made on bicycles, even though almost 40% of trips are two miles or less, which is a reasonable distance for cycling [1]. The primary deterrent to widespread cycling adoption is the prevailing perception of lack of safety [2].

Transportation researchers have dedicated significant efforts to understanding the comfort and stress experienced by cyclists over the years. Various classic assessment frameworks, such as Bicycle Level of Service (BLOS) [3] and Level of Traffic Stress (LTS) [4], have been established to quantify comfort or stress based on a comprehensive range of factors, including bike lane width, traffic volume, speed, and road surface quality. While these frameworks provide valuable insights, they endeavor to exhaustively enumerate and incorporate various contextual factors to derive the stress level, which is impractical to implement in real-world scenarios.

Alternatively, researchers have explored methods that involve self-report surveys [7, 9], where cyclists provide feedback on their perceived stress levels after each ride. While these surveys attempt to capture cyclists' experiences, they often fail to capture the "in the moment" stress cyclists face during their journeys due to the retrospective nature of self-report surveys, which rely on cyclists' memory and perception of stress after completing their rides. Moreover, the delay between the cycling experience and the survey response may lead to recall biases or inaccuracies in reporting, and such surveys can incur substantial costs [15].

Recent research has focused on studying cyclist stress through human physiology, using biomarkers such as heart rate [18, 5], and heart rate variability [20, 22]. Because of their reliance on specialized sensors that are not widely accessible, these methods are best suited for small-scale studies. Although heart rate can be monitored with common wearables, their limited market penetration in the US [24] suggests accessibility challenges for many cyclists. As a result, the practical use of physiological techniques in studying cyclist stress remains limited.

Cyclist stress is widely understood to be significantly impacted by traffic and environmental factors, a well-documented phenomenon in civil engineering research. Meanwhile, ongoing research in the field of mobile devices emphasizes the potential for using a variety of smartphone sensors to collect such contextual data [25, 26]. Inspired by these insights, CyclistAI, a novel smartphone-based cyclist stress assessment model that leverages context sensing, is developed. The model comprises a combination of a Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM). Training this model is straightforward with access to a sizable, annotated dataset. However, obtaining a well annotated dataset poses significant challenges in this case due to safety concerns. Cyclists may need to continuously report their perceived stress as labels for smartphone sensor readings, potentially diverting their attention and posing safety hazards. To address this issue, creating the annotated dataset using a bike simulator is proposed to train the stress assessment model. However, it's important to note that a model trained with simulation data may not be directly applicable to real-world cycling stress assessment due to differences between controlled lab environments and actual physical streets, leading to variations in stress perceptions. To bridge this gap, Cross-domain contrastive learning framework for domain adaptation is leveraged.

To evaluate CyclistAI, a prototype is created, and in-field testing is conducted in a real-world environment. The evaluation results indicate that CyclistAI achieves an average accuracy of 89%. It consistently performs well across diverse traffic and environmental conditions as a pioneering cyclist stress assessment model that relies solely on smartphone sensor readings.

The ultimate objective is to determine the stress distribution on roads and streets within a specific geographic area (such as a city, neighborhood, or region) by aggregating the assessment results from a group of proprietary cyclists' smartphone sensor readings as they bike through that area. As an illustration, in Figure 12, a visualized heat map of stress distribution over an area of interest is provided. It represents an aggregated result of the stress assessment from all participants. It will enable cyclists to select routes with low stress levels for their commute. Such a map will additionally benefit city planners, enabling them to identify areas requiring cycling infrastructure improvements. By analyzing these maps, city planners can prioritize locations for implementing measures such as adding dedicated bike lanes, improving road conditions, or reducing traffic congestion.

Chapter 2: RELATED WORK

The existing methods for assessing cyclist stress can be broadly classified into the following three categories:

Stress-rating frameworks: These methods quantify the cyclist stress by evaluating various traffic and environmental conditions [4, 39, 40], utilizing parameters such as bike lane width, traffic volume, road surface quality, and proximity to intersections. Bicycle Level of Service (BLOS) [3] employs a linear regression model with over ten parameters for estimating cyclist satisfaction. Similarly, Level of Traffic Stress (LTS) [4] proposes a function to model cyclists' stress levels. These methods establish the relationship between cyclist stress and environmental conditions through mathematical formulas, aiming to capture nuanced interactions between different factors. Recent advancements in research have explored both implicit and explicit relationships using artificial intelligence techniques [6, 11]. For instance, Huertas et al. [10] develop a stress classification model using unsupervised clustering and multinomial logistic regression. Professional equipment is typically used to measure road parameters in these models, and assessments are limited to the measured route.

Psychophysiology-based methods: Cyclist stress is assessed through physiological data analysis, including heart rate [5, 22], electrodermal activity [18], galvanic skin response [21], and respiration rate [16]. These methods offer direct insights into cyclists' psycho-physiological states. However, their widespread use is hindered by the reliance on costly research-grade sensors. While consumer-grade devices like smartwatches can monitor heart rate, their US adoption rate is only approximately 11% [24], particularly low among underserved populations. Thus, despite providing detailed information, practical application in large-scale cyclist stress assessments is limited by sensor availability and adoption rates.

Cognitive workload during cycling can be assessed using eye-tracking technology or physiological measures like pupillometry to gauge mental effort and information processing demands [42]. Various scales and instruments, such as the State-Trait Anxiety Inventory (STAI) [44], measure anxiety levels in cyclists. While these methods offer insights into cyclist stress, they often require specialized training for accurate interpretation, and the necessary tools may not be readily available.

Self-reported stress assessment: Historically, cyclist stress levels have been assessed using self-report methodologies [7, 9]. These methods typically involve direct surveys [8] or participants rating their stress levels after viewing videotaped depictions of diverse on-street traffic scenarios. However, despite their widespread use, these approaches can be resource-intensive and may fall short in capturing the real-time experience of cyclists [15]. Additionally, the retrospective nature of these methods may introduce biases and inaccuracies in reporting.

Chapter 3: BACKGROUND AND MEASUREMENT STUDY

3.1 Background

Over the past decade, studies have demonstrated that environmental and traffic conditions significantly influence a cyclist's stress levels. Factors such as road surface quality, traffic density, transportation infrastructure, air quality, illumination, and weather conditions all play a major role. Cyclists often face challenges such as navigating potholes, uneven roads, and debris on subpar road surfaces, which can lead to increased stress levels. Furthermore, they must always remain vigilant for potential hazards, further contributing to their stress. Research in civil engineering has consistently shown that cyclists experience heightened stress when riding in heavy traffic, as they perceive a greater risk of accidents in congested areas. Fortunately, many smartphone sensors are now capable of capturing the vast array of contextual information mentioned above. Previous studies [25] have utilized smartphone accelerometers to detect road irregularities. These studies revealed a correlation between noise levels, audio events captured by smartphone microphones, and nearby traffic volumes [27]. Furthermore, smartphones equipped with ambient light and GPS sensors can effectively monitor street illumination conditions [28].

These combined pieces of evidence encourage us to leverage smartphone sensors to capture contextual information about cyclists' surroundings while they ride through urban roads, with the final objective of assessing their stress levels.

3.2 Measurement Study

A preliminary measurement study is carried out to examine the feasibility of this concept.

3.2.1 Measurement Setup

A smartphone app was developed for collecting sensor data to record the cyclist's environmental context, including road conditions and traffic noise, utilizing two types of sensors: IMU and microphone. Six participants were enlisted to cycle around the campus, with their smartphones affixed to the handlebars. Bicycle maneuvers were indicated by IMU measurements; however, these were heavily influenced by road conditions. At regular 5-second intervals, participants were prompted by the smartphone app to verbally report their perceived stress levels, categorized as low, medium, or high. The audio captured by the microphone was

subsequently transcribed by human auditors into labels corresponding to the sensor readings. To address the potential interference of verbal reports with the microphone's recording of traffic conditions, a voice activity detection algorithm [59] was employed to identify and remove the stress report segments from the recorded audio tracks during post-processing. Subsequently, statistical analysis was conducted to explore the potential relationship between instantaneous sensor readings and the associated cyclist-reported stress levels. It's important to note that the entire study in this work received approval from the Institutional Review Board (IRB).

3.2.2 Statistical Analysis

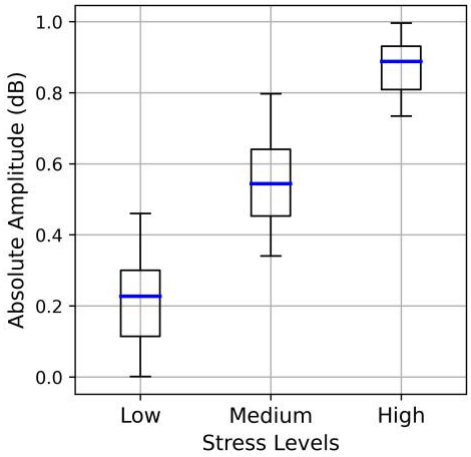
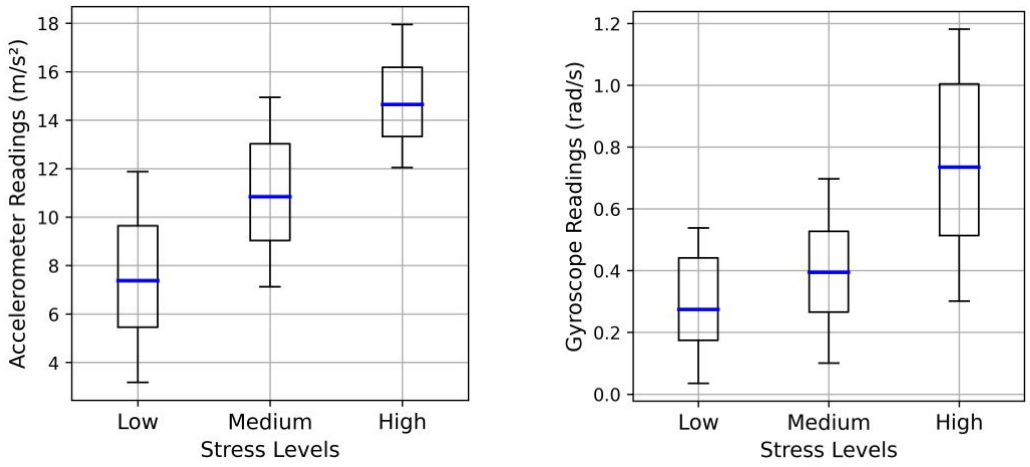


Figure 1: Boxplot of absolute amplitude under different stress levels.

Figure 1 depicts the absolute amplitude across various stress levels within a randomly selected audio sample. Notably, as stress levels escalate, there is a corresponding rise in amplitude. This suggests that in noisy and busy environments, cyclists are more likely to experience elevated stress levels.

Figure 2 demonstrates the distribution of IMU readings collected from the accelerometer and gyroscope across three stress levels. The data have been aggregated from measurements across all axes (x, y, z) to create scalar values, representing total acceleration and angular velocity. The distribution for each stress level is based on 360 five-second signal segments from all six participants. There are clear correlations: increased reported stress levels are typically correlated with increased acceleration and angular velocity in the measurements.



(a) Acceleration from accelerometer (b) Angular velocity from gyroscope

Figure 2: Boxplot of IMU readings

The findings of the measurement study show encouraging results. There is a notable correlation between smartphone sensor readings and the stress levels experienced by cyclists.

During periods of high stress, such as navigating through heavy traffic or challenging terrain, audio recordings may capture elevated background noise levels, including honking cars or loud engine revving. These audio cues reflect the heightened environmental stimuli and increased cognitive load experienced by the cyclist. On the other hand, in low-stress environments, such as quiet residential streets or dedicated bike paths, audio recordings may exhibit minimal background noise. This correlation suggests that ambient noise levels can indicate the level of stress experienced by the cyclist during their ride.

Different stress levels also exhibit distinctive patterns in the accelerometer readings. In high-stress situations, characterized by heavy traffic, cyclists experience frequent starts and stops, leading to rapid fluctuations in acceleration. Conversely, during low-stress periods, such as in light traffic conditions, cyclists enjoy smoother rides with consistent acceleration and minimal disturbances.

Similar to accelerometer readings, gyroscope data reflects varying stress levels encountered by cyclists. In high-stress scenarios, comprising factors like high traffic, poor road conditions and adverse weather, cyclists must make frequent adjustments to maintain balance, resulting in significant fluctuations in gyroscope readings. In contrast, during low-stress conditions, such as

on well-paved roads with minimal obstacles, gyroscope adjustments are infrequent, leading to stable readings.

Figure 3: Bike Simulator

3.2.3 Data Collection in Simulation Environment

Encouraged by these observations, the goal is to develop an advanced deep learning model that can effectively translate the sensor readings into cyclist stress levels automatically.

In order to develop a model that can adapt to different traffic and infrastructural situations in city streets, an extensive labeled dataset with annotated stress levels is essential. However, there are a few safety issues with data annotation. Continuous reporting of perceived stress by cyclists which acts as labels for the smartphone's sensor readings may divert their attention and put them in danger as they bike in a busy metropolitan area. This is more likely in scenarios when there is a lot of traffic. This approach is useful for collecting a small quantity of data under controlled conditions, but it is not sufficient to obtain the large dataset needed to effectively train a deep learning model. To address this issue, annotated smartphone data is collected from an immersive simulator environment as shown in Figure 3.

Chapter 4: BASIC CYCLIST STRESS ASSESSMENT MODEL

In order to address the complexities of cyclist stress assessment, the basic model proposed in this project relies on a fusion of Convolutional Neural Networks and Long Short-Term Memory Networks. To fully comprehend the architectural intricacies and theoretical underpinnings of this model, it is imperative to delve into the realm of deep learning.

4.1 Deep Learning

Deep learning [31], a subset of machine learning, has revolutionized various fields with its ability to process complex data and extract meaningful patterns for prediction and decision-making. In recent years, deep learning techniques have gained widespread popularity and have been applied across diverse domains, including computer vision, natural language processing, healthcare, finance, and autonomous systems.

At its core, deep learning seeks to mimic the human brain's neural networks to process and interpret data. By utilizing interconnected layers of neurons, known as artificial neural networks, deep learning models can automatically learn intricate representations of input data, enabling them to perform tasks such as pattern recognition, classification, regression, and anomaly detection.

One of the key advantages of deep learning lies in its capability to handle large volumes of unstructured data efficiently. Unlike traditional machine learning algorithms that rely on handcrafted features, deep learning models can automatically learn hierarchical representations of data directly from raw inputs. This ability is particularly advantageous in tasks where manually designing features is impractical or infeasible, such as image and speech recognition.

4.1.1 Artificial Neural Networks (ANNs)

Artificial Neural Networks are computational models inspired by the structure and function of the human brain [14]. They consist of multiple layers of interconnected neurons, with each neuron serving as the fundamental processing unit. These networks process information through a series of weighted connections, where inputs are received, combined with corresponding weights, summed, and then passed through an activation function to produce an output. ANNs typically comprise an input layer for receiving initial data, one or more hidden layers responsible

for learning representations of the data, and an output layer for producing final predictions or classifications. This architectural layout is illustrated in Figure 4.

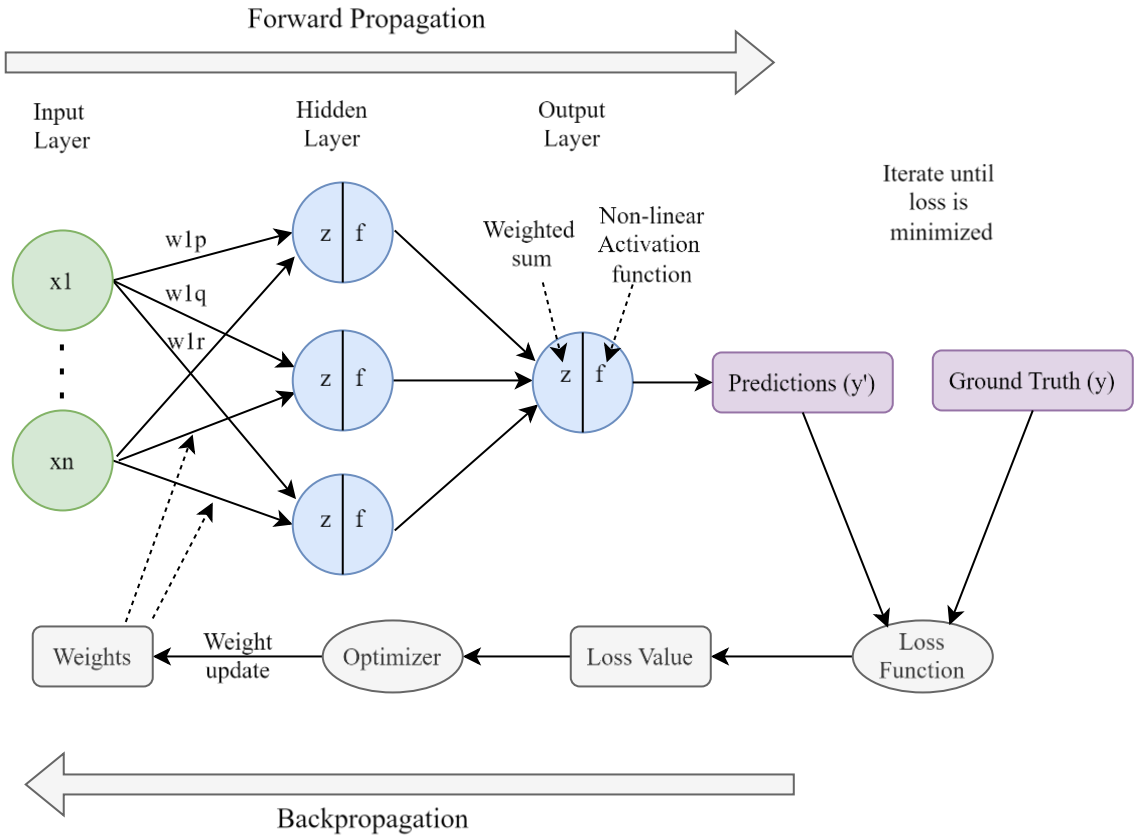


Figure 4: Artificial Neural Network Architecture

Training an ANN involves two primary processes:

1. Forward Propagation: This process entails sequentially computing the output of each neuron in each layer until reaching the output layer. Mathematically, the output $z_j^{(l)}$ of a neuron j in layer l can be represented as:

$$z_j^{(l)} = \sum_{i=1}^{n} w_{ij}^{(l)} x_i^{(l-1)} + b_j^{(l)}$$

where ${w_{ij}}^{(l)}$ denotes the weight associated with the connection between neuron *i* in layer *l-1* and neuron *j* in layer *l*, ${x_i}^{(l-1)}$ represents the output of neuron *i* in layer *l-1*, and ${b_j}^{(l)}$ is the bias term for neuron *j* in layer *l*. The weights of the network are initialized randomly at the beginning. The output ${z_j}^{(l)}$ then passes through an activation function, $f(z)$ introducing non-linearity.

$$a_j^{(l)} = f(z_j^{(l)})$$

Common activation functions include sigmoid, ReLU, and tanh. Each neuron in subsequent layers computes its output, which serves as input for neurons in the next layer. The output of the final layer represents the network's predictions.

2. Backpropagation: This process involves refining the learnable parameters (i.e. weights) of the previous layers through iterative adjustments. It minimizes prediction errors by updating weights based on computed gradients. For each input sample, a forward pass is performed through the network, computing the output of each neuron in each layer using the current weights and activation functions. After obtaining the predicted output of the network, the loss between the predicted output y_{pred} and the true output y_{true} is calculated using a chosen loss function. L = loss (y_{true}, y_{pred}).

   Now, the weights are updated using gradient descent, iterating through the network weights until convergence or a stopping criterion is met. The weight update rule is as follows:

$$w_{ij}^{(l)} = w_{ij}^{(l)} - \alpha \frac{\partial L}{\partial w_{ij}^{(l)}}$$

   where α is the learning rate which controls the optimization step size.

   The gradient of the loss function with respect to the weight $w_{ij}^{(l)}$ connecting neuron *i* in layer *l-1* to neuron *j* in layer *l* can be expressed using the chain rule as:

$$\frac{\partial L}{\partial w_{ij}^{(l)}} = \frac{\partial L}{\partial a_j^{(l)}} . \frac{\partial a_j^{(l)}}{\partial z_j^{(l)}} . \frac{\partial z_j^{(l)}}{\partial w_{ij}^{(l)}}$$

where $a_j^{(l)}$ is the activation of neuron *j* in layer *l*, and $z_j^{(l)}$ is the weighted sum of inputs of neuron *j* in layer *l*. This weight update rule adjusts the weights based on the gradient of the loss function with respect to each weight, thus guiding the optimization process to minimize the error.

By iteratively employing both forward propagation and backpropagation, the network's weights are progressively adjusted to minimize the loss function, thereby enhancing the network's performance on the designated task.

4.1.2 Long Short-Term Memory (LSTM)

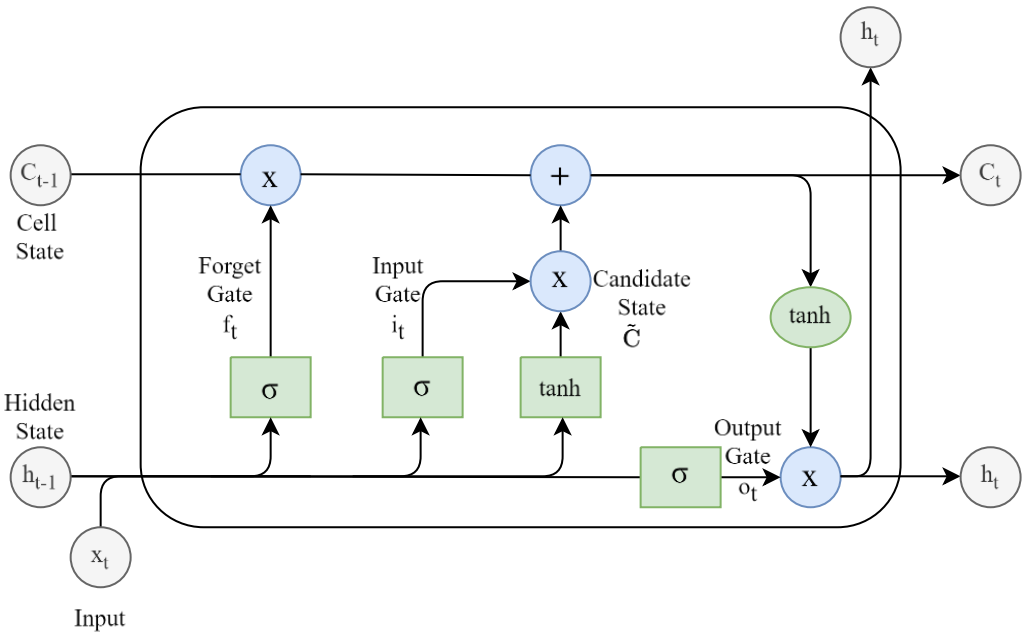


Figure 5: Long Short-Term Memory Architecture

Long Short-Term Memory networks [32] constitute a pivotal advancement in the domain of sequential or time series data analysis, adept at capturing long-range dependencies and mitigating the vanishing gradient problem [60]. At the core of LSTM architecture lie LSTM cells, organized sequentially. Each LSTM cell comprises critical components, including the cell state, three multiplicative gates-input, forget, and output gates-constructed using neural network layers. These gates, along with activation functions like hyperbolic tangent or sigmoid, regulate the flow of information into, out of, and within the cell state, facilitating selective storage or discard of information over extended sequences. The cell state (C_t) represents the long-term

memory of the network, while the hidden state (h_t) captures relevant information from the input sequence at a given time step.

LSTMs process sequential or time series data, where x_t represents the input at time step *t*. Mathematically, LSTM operations are encapsulated in equations governing state transitions as shown below:

- The forget gate (f_t) determines which information from the previous cell state (C_{t-1}) should be discarded. It computes $f_t = \sigma\ (W_f\ .\ [h_{t-1},\ x_t] + b_f)$, where σ is the sigmoid activation function, W_f is the weight matrix for the forget gate, and b_f is the bias vector.

- The input gate (i_t) decides which new information ($\tilde{C}_t$) should be added to the cell state. It computes $i_t = \sigma\ (W_i\ .\ [h_{t-1},\ x_t] + b_i)$ where W_i is the weight matrix for the input gate, and b_i is the bias vector.

- The candidate cell state ($\tilde{C}_t$) represents the new information proposed to be added to the cell state. It computes $\tilde{C}_t = \sigma\ (W_C\ .\ [h_{t-1},\ x_t] + b_C)$, where W_C is the weight matrix for the candidate cell state, and b_C is the bias vector.

- The output gate (o_t) controls the flow of information from the current cell state to the hidden state (h_t). It computes $o_t = \sigma\ (W_o\ .\ [h_{t-1},\ x_t] + b_o)$, where W_o is the weight matrix for the output gate, and b_o is the bias vector.

- The cell state (C_t) and hidden state (h_t) are updated using equations $C_t = f_t\ .\ C_{t-1} + i_t\ .\ \tilde{C}_t$ and $h_t = o_t\ .\ \tanh\ (C_t)$ respectively. This hidden state serves as the output for time step *t*.

These equations detail how information is processed and updated within the LSTM unit at each time step. Through this intricate interplay of gating mechanisms and memory management, LSTMs excel in various sequential learning tasks, from natural language processing to time series forecasting, cementing their status as a cornerstone in deep learning architectures.

4.1.3 Convolutional Neural Networks (CNN)

CNNs [35] are a type of artificial neural network particularly well-suited for processing structured grid-like data, such as images or time series data. They have been highly successful in various tasks, including image recognition, object detection, and natural language processing. When it comes to sensor time series data, CNNs can be adapted to effectively extract features and patterns, making them suitable for a wide range of applications, including time series analysis, signal processing, and natural language processing.

4.1.4 1-Dimensional Convolutional Neural Networks (1D CNN)

1D CNNs [36] are tailored to capture patterns and temporal dependencies within one-dimensional sequential or time-series data. The main components of a 1D CNN are presented below:

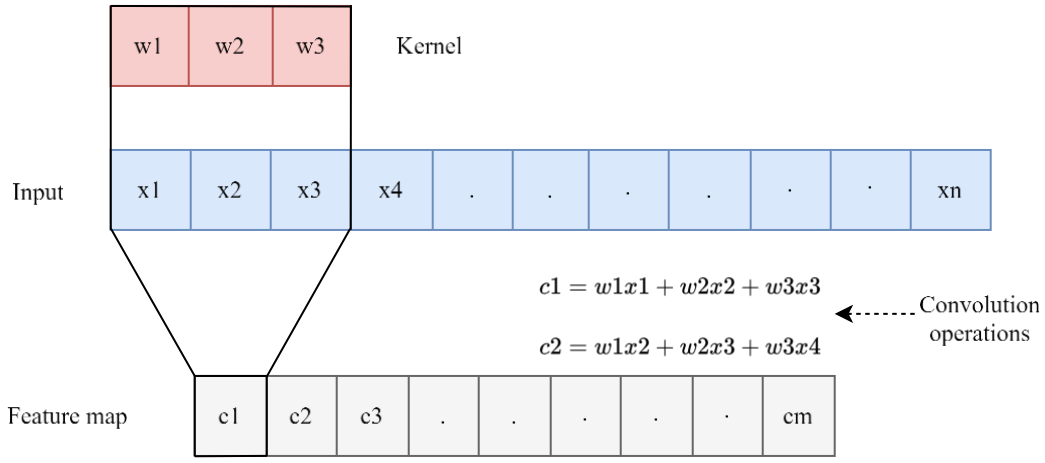


Figure 6: 1D Convolution operation (kernel size=3, number of kernels=1)

1. Input Layer: The initial layer of the network receives the one-dimensional sequential data, which could be signals, time series, audio, or text.

2. Convolutional Layers: These layers apply one-dimensional convolutions to the input data as illustrated in Figure 6. Each convolutional layer consists of multiple filters or kernels that convolve over the input sequence to extract local patterns and features. The output of each convolution operation forms a feature map, which represents the presence of certain features at different positions in the input sequence.

3. Activation Layers: Activation functions [19] like ReLU, hyperbolic tangent are applied element-wise to the feature maps obtained from the convolutional layers. This introduces non-linearities into the network, allowing it to learn complex relationships and representations from the input data.
4. Pooling Layers: Pooling layers down sample the feature maps by summarizing the information in local regions. Max pooling and average pooling are common pooling operations used in 1D CNNs. Pooling helps reduce the spatial dimensions of the feature maps while preserving the most important information, making the network more computationally efficient and robust to variations in the input data.

5. Flattening: Following the convolutional and pooling layers, the resulting feature maps are flattened into a vector. This vector serves as the input to the subsequent fully connected layers.

6. Fully Connected Layers: These layers process the flattened feature vector to make predictions or classifications. They learn complex combinations of features extracted by the convolutional layers and make decisions based on those features. These layers are often followed by activation functions.

7. Output Layer: The output layer produces the final predictions or classifications. Depending on the task, it might consist of a single neuron for regression or multiple neurons with SoftMax activation for classification.

The complete architecture of 1D CNN is presented in Figure 7.

4.2 Basic Stress Assessment Model Architecture

Having introduced the fundamental concepts of deep learning, it is essential to delve deeper into the specifics of the basic model. Afterwards, the limitations of the initial model are addressed, which prompts the creation of a more advanced model in the next chapter. The basic model is crucial as it lays the groundwork for the entire design.

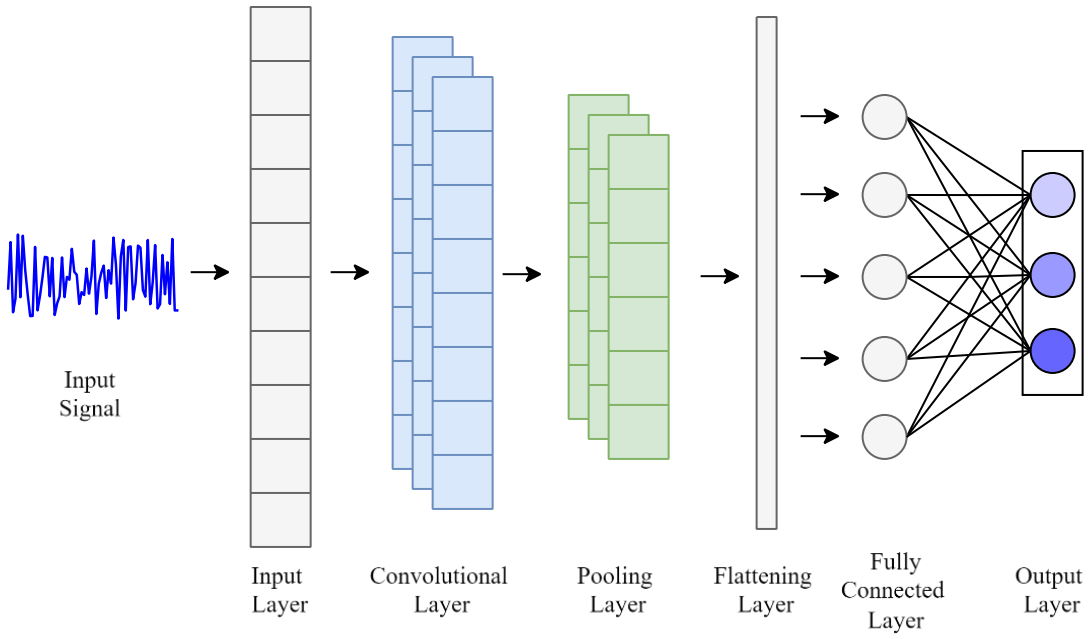


Figure 7: 1D CNN Architecture

4.2.1 Feature Encoders

The feature encoders consist of a combination of CNN and LSTM architectures. This fusion of 1D CNN [36] and LSTM networks is particularly well-suited for processing both audio and IMU data due to their unique characteristics.

1D convolutional layers excel at capturing local patterns within sequential data, making them highly effective for extracting features from the temporal structure present in both audio and IMU sensor readings. For instance, in audio data, CNNs can capture short-term audio features such as pitch, timbre, and temporal dynamics, by convolving over small segments of the audio waveform to identify patterns indicative of specific auditory characteristics. This allows CNNs to discern variations in sound intensity, frequency, and texture, enabling the model to detect environmental cues that may contribute to stress in bicyclists. Similarly, in IMU data, CNNs can identify patterns corresponding to specific movements or gestures over time, providing insights into the physical activities and interactions experienced by the cyclist.

In contrast, LSTM networks are adept at modeling long-range dependencies and temporal dynamics within sequential data. In the context of bicyclist stress detection, LSTM can be particularly useful for analyzing audio data to capture nuances in environmental sounds that unfold over time. For example, LSTM can recognize patterns in the duration, frequency, and

intensity of sounds such as traffic noise, sirens, or construction activity, which may indicate stressful situations for bicyclists. Similarly, in IMU data, LSTM can capture complex movement patterns or gestures that evolve gradually, providing insights into the cyclist's physical exertion, balance, and interaction with the environment.

By combining CNNs and LSTMs, the model leverages the complementary strengths of both architectures, allowing it to effectively capture and represent the intricate temporal patterns present in both audio and IMU data. This synergistic approach enhances the model's ability to extract discriminative features and achieve superior performance in tasks such as stress detection, where accurate representation of temporal dynamics is crucial.

In this research, the IMU data is derived from a selection of six axes, specifically three axes from the accelerometer and three axes from the gyroscope, while the audio data remains unidimensional, representing amplitude variations. To handle the heterogeneity inherent in these data modalities, separate feature encoders are utilized to extract unimodal features. This approach ensures that distinct characteristics of IMU and audio data are effectively captured. Figure 8 demonstrates the architecture of audio and IMU feature encoders.

Consider a dataset of *N* multimodal data samples, $x = \{x^i | i = 1, \dots, N\}$ where each sample $x^i = \{x_u^i, x_a^i\}$ contains IMU data x_u^i and audio data x_a^i. Each x^i is individually processed through unimodal feature encoders $f_{enc.u}(\cdot)$ for IMU data and $f_{enc.a}(\cdot)$ for audio data, generating separate representation vectors. Subsequently, a flattening operation is applied to both IMU and audio data representations, resulting in one-dimensional vectors denoted as h_u^i and h_a^i, respectively. Despite their one-dimensional nature, these vectors may differ in the number of hidden features they contain.

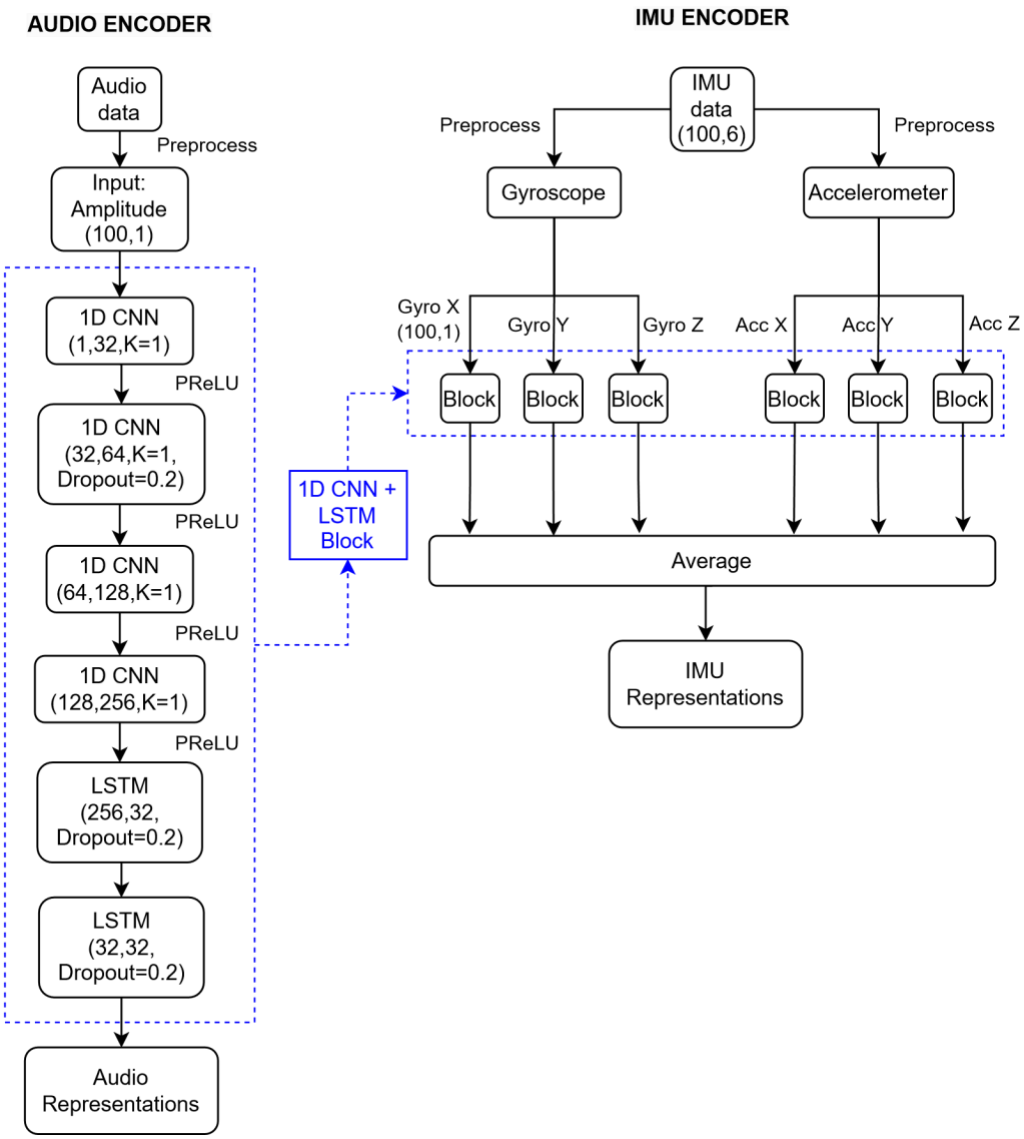


Figure 8: Feature Encoder Architecture (batch size=100)

4.2.2 Stress Classifier

Two fully connected layers are integrated at the end of the feature encoders to serve as the cyclist stress classifier, which then undergoes supervised learning. The features obtained from various sensors are merged, denoted as c^i = Concatenate (h_u^i, h_a^i), and subsequently flattened and input into the classifier which is trained using the labeled source dataset.

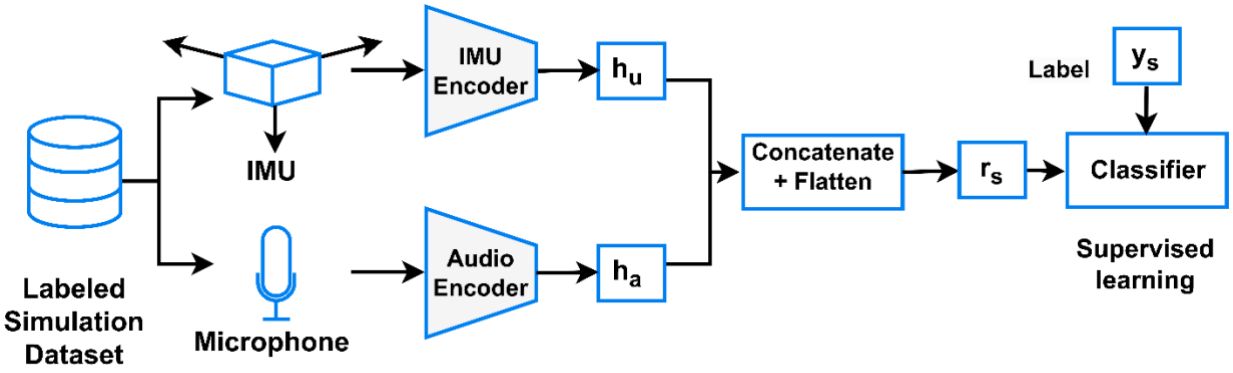


Figure 9: Basic Cyclist Stress Model Architecture

4.2.3 Training

Figure 9 illustrates the architecture and the training process of the basic model. The labeled source dataset is collected from a bicycle simulator, as mentioned in the measurement study. The simulator provides a controlled environment where various cycling scenarios, including urban and rural road types, different weather conditions, and variations in traffic volume, can be simulated. While cycling, participants will interact with simulated environments and verbally report their perceived stress. These verbal accounts will be used as annotations for the sensor data. The basic model is trained using gradient descent [23] until the loss is minimized.

4.2.4 Limitations of the Basic Model

We evaluate the basic model on testing datasets collected from both simulation and real-world environments. Our observations reveal high accuracy on simulated data under various traffic conditions, contrasted with moderate accuracy on real-world datasets, as depicted in Figure 10. While the model demonstrates capability in recognizing patterns and making predictions based on provided features, it falls short of achieving high accuracy levels, especially in high traffic conditions. The primary reason for the low accuracy of the basic model is the significant domain gap between simulator and real-world data. Despite efforts to simulate various cycling scenarios, the controlled environment of the simulator may not fully capture real-world complexities such as environmental conditions, road surfaces, traffic interactions, and cyclist behaviors. For instance, in a simulator environment, a cyclist might experience an overall lower stress level compared to the stress experienced while riding on real streets due to

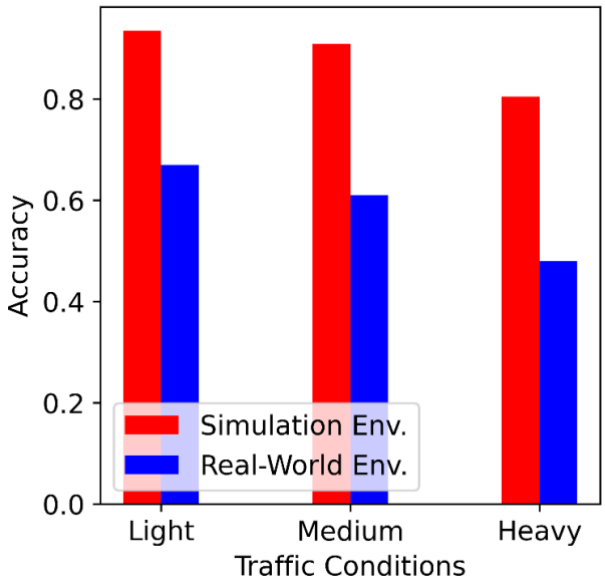


Figure 10: Basic Model Evaluation

the controlled environment's inability to replicate real-time traffic dynamics, unpredictable weather conditions, and interactions with other road users. This discrepancy hampers the model's ability to adapt effectively to real-world scenarios, resulting in variations in stress perceptions. Consequently, concerns arise regarding the model's reliability and applicability in real-world settings, where it may encounter scenarios and conditions unrepresented in the simulator data. Addressing the challenge of domain shift is crucial for enhancing the model's generalization capabilities and ensuring its effectiveness in real-world applications.

Chapter 5: ADVANCED MODEL: CyclistAI

5.1 Introduction

As outlined in the limitations of the basic model, a model trained solely on simulation data cannot be directly applied to real-world cycling stress assessment due to inherent distinctions between controlled laboratory environments and real-world streets, which can lead to variations in stress perceptions. Therefore, to enhance performance, a comprehensive dataset encompassing both simulated and real-world cycling data is utilized for training. Acquiring simulation data is straightforward, as detailed in the measurement setup through a controlled bike simulation environment. Participants ride bikes with smartphones affixed to the handles, equipped with an app that records sensor readings. They verbally report their perceived stress level (low, medium or high) every 5 seconds, which is also recorded by the app and serves as labels for the simulated data.

In the real-world data acquisition phase, using the same verbal reporting approach could divert the bicyclists' attention, posing potential safety risks. Hence, participants are instructed to simply bike around the campus without verbally reporting their stress levels. Only unlabeled sensor data is collected via the smartphone app. It is important to note that collecting unlabeled data in real-world settings demands minimal effort from cyclists.

Given the disparity between simulator and real-world data, it's clear that directly merging labeled and unlabeled data lacks meaningful coherence. To address this, the advanced model adopts a novel approach at integrating the two modalities (audio and IMU) from different domains and translating these contextual measures into cyclist stress using domain adaptation [30]. Domain adaptation enables a deep learning model trained in a source domain (i.e., data collected in a lab environment) to adapt to a different but related target domain (i.e., real-world data). Building upon the foundational principles of the basic model, the advanced model, CyclistAI, is now introduced.

5.2 Feature Encoders

In the advanced model, the feature encoders remain the same as those used in the basic model. This ensures consistency in the initial processing stages, allowing for a seamless

transition between the two models while focusing on enhancing the model's performance through domain adaptation and integration of real-world data.

5.3 Projection Networks

Projection networks [37] are employed to standardize the output features from different modalities to have uniform dimensions. This strategy enhances the model's ability to process and analyze the diverse information sources, facilitating accurate representation and interpretation of the temporal dynamics present in both audio and IMU data.

To address the differences in the number of hidden features between IMU and audio data, two multilayer projection networks composed of 1D convolutional layers, denoted as $g_u(\cdot)$ and $g_a(\cdot)$, are utilized. These networks align the IMU and audio features to a common dimension. Additionally, both sets of features undergo normalization to maintain consistent dimensions, resulting in representations z_u^i and z_a^i that are positioned on a unit hypersphere. This harmonization process ensures that the diverse inputs from IMU and audio sources are appropriately prepared for subsequent operations.

5.4 Contrastive Learning for Domain Adaptation

To bridge the simulation-reality gap discussed above, it is framed as a domain adaptation problem [43] and addressed through contrastive learning. The feature encoders are denoted as $f_{\theta enc}(\cdot)$ with network parameters θ. The source dataset $D_s = \{(x_s^i, y_s^i)\}$ represents simulation data, where x_s^i is the sensor data of the *i*-th sample with y_s^i as its label, denoting a discrete cyclist stress level from low to high. $D_s = \{(x_t^i)\}$ is the unlabeled target real-world dataset. The goal of domain adaptation is to obtain domain-invariant features using the target domain data x_t^i as input, employing $f_{\theta enc}(\cdot)$ trained on $D_s \cup D_T$.

5.4.1 Cross-Domain Contrastive Learning

The goal of domain adaptation is to bring the feature distributions in the source and target domains into synchrony. Cross-domain contrastive learning aims to do this by combining features from samples belonging to the same class and separating features from samples belonging to distinct classes, irrespective of their domain of origin. The cross-domain contrastive learning loss function is now presented in order to accomplish this goal.

The first step involves fusing features from different sensor modalities, where $r^i = Concatenate(z_u^i, z_a^i)$ represents the fusion process. In the target domain, the fused feature r_t^i is termed as the anchor, while the fused feature r_s^p in the source domain with an identical label is referred to as a positive feature. Other samples in the source domain contribute to negative features r_s^n. The cross-domain contrastive loss is then defined as:

$$L_{CDC}^{t,i} = -\frac{1}{|P_s(i)|}\sum_{p\in P_s(i)} log\frac{exp\,(r_t^i.r_s^p/\tau)}{\sum_{n\in D_s} exp\,(r_t^i.r_s^n/\tau)}$$

Equation 1

where *i*, *p*, and *n* denote the indices of fused features from different samples, $D_s \equiv \{1,2,\dots,N_s\}$ represents the set of source samples in a mini-batch, $P_s(i) \equiv \{p \in D_s : y_s^p = \hat{y}_t^i\}$ is the set of positive samples from the source domain sharing the same label with the anchor, and τ serves as a temperature factor modulating the influence of various samples [38]. This loss function enforces the intra-class distance to be smaller than the inter-class distance for samples from different domains, thus mitigating domain shift.

Due to the lack of access to labels for target samples, estimated pseudo labels $\hat{y}_t^i$ are used to generate pairs. In Equation 1, samples from the target domain are treated as anchor points. $L_{CDC}^{s,i}$ can be computed similarly, treating source samples as anchor points and setting, $P_t(i) \equiv \{p \in D_t : \hat{y}_t^p = y_s^i\}$, where $D_t \equiv \{1,2,\dots,N_t\}$ is the set of target samples in a mini-batch.

By merging $L_{CDC}^{s,i}$ with $L_{CDC}^{t,i}$, the cross-domain contrastive loss [54] is formulated as:

$$L_{CDC} = \sum_{i=1}^{N_s} L_{CDC}^{s,i} + \sum_{i=1}^{N_t} L_{CDC}^{t,i}$$

Equation 2

This loss facilitates bidirectional feature alignment by incorporating anchors from both domains, resulting in enhanced performance. The ultimate objective function for training is:

$$\underset{\theta}{minimize}\ L_{CDC}(\theta\ ;\ D_s, D_T)$$

Equation 3

where θ represents the parameters subject to optimization.

5.4.2 Pseudo Label Assignment

Due to safety concerns and the unavailability of cyclist stress level data in the real-world traffic dataset, pseudo labels are generated using K-means clustering [41] for pairing in cross-domain contrastive learning [49, 50, 12, 13]. Unlike conventional K-means clustering, where cluster centroids are randomly selected, cluster centroids in the target domain are initialized to match those in the source domain. This initialization is based on the premise that source class prototypes serve as approximations of target class prototypes, owing to shared semantic information. As training progresses, this approximation is refined as samples within the same category align through cross-domain contrastive learning. Initially, source class prototypes are established by calculating centroids from source samples within each stress level.

$$O_t^l \leftarrow O_s^l = E_{i \sim D_S,\ y_s^i = l} r_s^i$$

Subsequently, to assign a pseudo label to each target sample, cosine similarity is employed to measure the distance between the target sample r_t^i and the *l*-th cluster center O_t^l. Following the clustering process, each target sample x_t^i is associated with a pseudo label $\hat{y}_t^i$. To mitigate noise within the target pseudo labels, samples exhibiting ambiguity are excluded. Specifically, a target sample is discarded if the cosine similarity between its feature and its assigned cluster center falls below an empirical threshold *d*.

5.5 Stress Classifier

To build the cyclist stress model, two fully connected layers are attached at the end of feature encoders, and supervised learning is performed with using a small labeled real-world dataset D_r. The labels denoted as y_r, are utilized along with the feature representations as inputs, as delineated in Algorithm 1.

The classifier clf is responsible for predicting the stress level of cyclists based on the extracted features. To optimize the classifier's parameters ($\theta_{classifier}$) for accurate stress predictions, the training process employs the cross-entropy loss function [45], a commonly used

metric in classification tasks. Cross-entropy loss quantifies the disparity between predicted probability distributions and true class labels, providing a measure of how well the classifier's predictions align with the ground truth. In the context of multiclass classification tasks, the cross-entropy loss equation is expressed as:

$$L(y, \hat{y}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} \log\left(\hat{y}_{ij}\right)$$

Equation 4

where *y* represents the true label, $\hat{y}$ is the predicted probability that sample *i* belongs to class *j*, *N* is the number of samples, and *C* is the number of classes. During training, the parameters of the classifier are iteratively updated using gradient descent, an optimization algorithm that adjusts the model parameters in the direction of the steepest descent of the loss function [23]. By minimizing the cross-entropy loss through gradient descent, the classifier learns to better differentiate between different stress levels, ultimately improving its performance in accurately classifying cyclists' stress levels.

Algorithm 1 Cyclist Stress Classifier Training

---

procedure CLASSIFIERTRAINING (D_r, $\theta_{classifier}$)
    Load trained encoders f_{audio}, f_{imu} with frozen weights
    Initialize classifier clf with random weights
    for $i = 1$ **to** N_r **do**
        $x_r^i, y_r^i \in D_r^i$
        $a_r^i, u_r^i \leftarrow$ extract_features(x_r^i)
        $h_a^i \leftarrow f_{audio}(a_r^i)$, $h_u^i \leftarrow f_{imu}(u_r^i)$
        $z^i \leftarrow flatten(Concatenate\left(h_a^i, h_u^i\right))$
        $\hat{y}^i \leftarrow clf\left(z^i\right)$
        Compute $L\left(y_r^i, \hat{y}^i\right)$ using Equation 4
        Update $\theta_{classifier}$ using gradient descent
    end for
end procedure

5.6 System Architecture

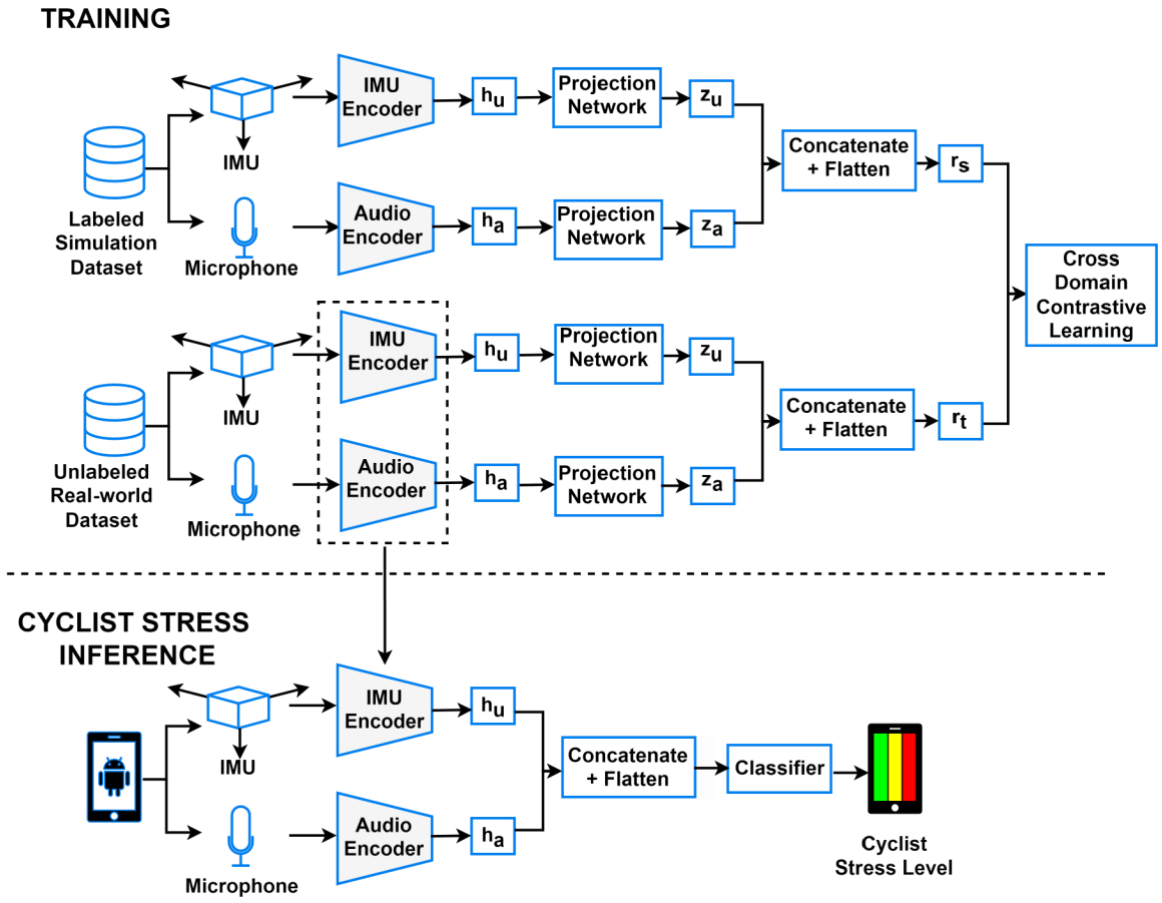


Figure 11: CyclistAI System Architecture

Figure 11 depicts the system architecture of CyclistAI. The initial phase involves training the feature encoders using a labeled dataset from a cycling simulator and an unlabeled dataset collected from the real-world environment. Domain adaptation is achieved through contrastive learning using Algorithm 2. Now, the feature encoders yield domain invariant representations, directly utilized for downstream classification tasks. This decision stems from the observation that the representation before the projection captures more informative features than the projection output [37]. Consequently, projection networks are discarded, streamlining the model architecture and boosting computational efficiency.

After the training phase, both the trained encoders and the trained classifier are exported to the ONNX format for deployment within the Android application in the final phase. During on-device inference, the IMU sensor data and microphone data are directed to the encoders. The resulting feature representations are then passed into the classifier. The classifier is seamlessly

integrated with the interface, which visually presents the bicyclist's stress level. In the experiment, the inference frequency has been set to one inference per 0.5 seconds. The inferred cyclist stress level is temporarily stored on the device and uploaded to the server when a stable network connection is detected. This stress is used to plot the cyclist stress heatmap as shown in Figure 12.

Algorithm 2 Cross-Domain Contrastive Learning (CDCL)

---

procedure CDCL ($D_S, D_T, num\_epochs, \theta_{audio}, \theta_{imu}, \theta_{projection}$)
    Initialize encoder f_{audio}, f_{imu} weights
    Initialize projection network g weights
    Initialize cluster centers with source class prototypes
    Initialize τ as the temperature factor
    for $epoch = 1$ **to** num_epochs **do**
        $L_{CDC} \leftarrow 0$
        for $i = 1$ **to** N_s **do**
            $a_s^i, u_s^i \leftarrow$ extract_features(x_s^i)
            $h_a^i \leftarrow f_{audio}(a_s^i), h_u^i \leftarrow f_{imu}(u_s^i)$
            $z_s^i \leftarrow flatten(Concatenate(h_a^i, h_u^i))$
            $r_s^i \leftarrow g(z_s^i)$
            Compute $L_{CDC}^{s,i}$ for r_s^i anchor using Equation 1
        end for
        for $i = 1$ **to** N_t **do**
            $a_t^i, u_t^i \leftarrow$ extract_features(x_t^i)
            $h_a^i \leftarrow f_{audio}(a_t^i), h_u^i \leftarrow f_{imu}(u_t^i)$
            $z_t^i \leftarrow flatten(Concatenate(h_a^i, h_u^i))$
            $r_t^i \leftarrow g(z_t^i)$
            Compute $L_{CDC}^{t,i}$ for r_t^i anchor using Equation 1
        end for
        Compute batch L_{CDC} using Equation 2
        Update parameters $\theta_{audio}, \theta_{imu}, \theta_{projection}$ using Equation 3
    end for
end procedure

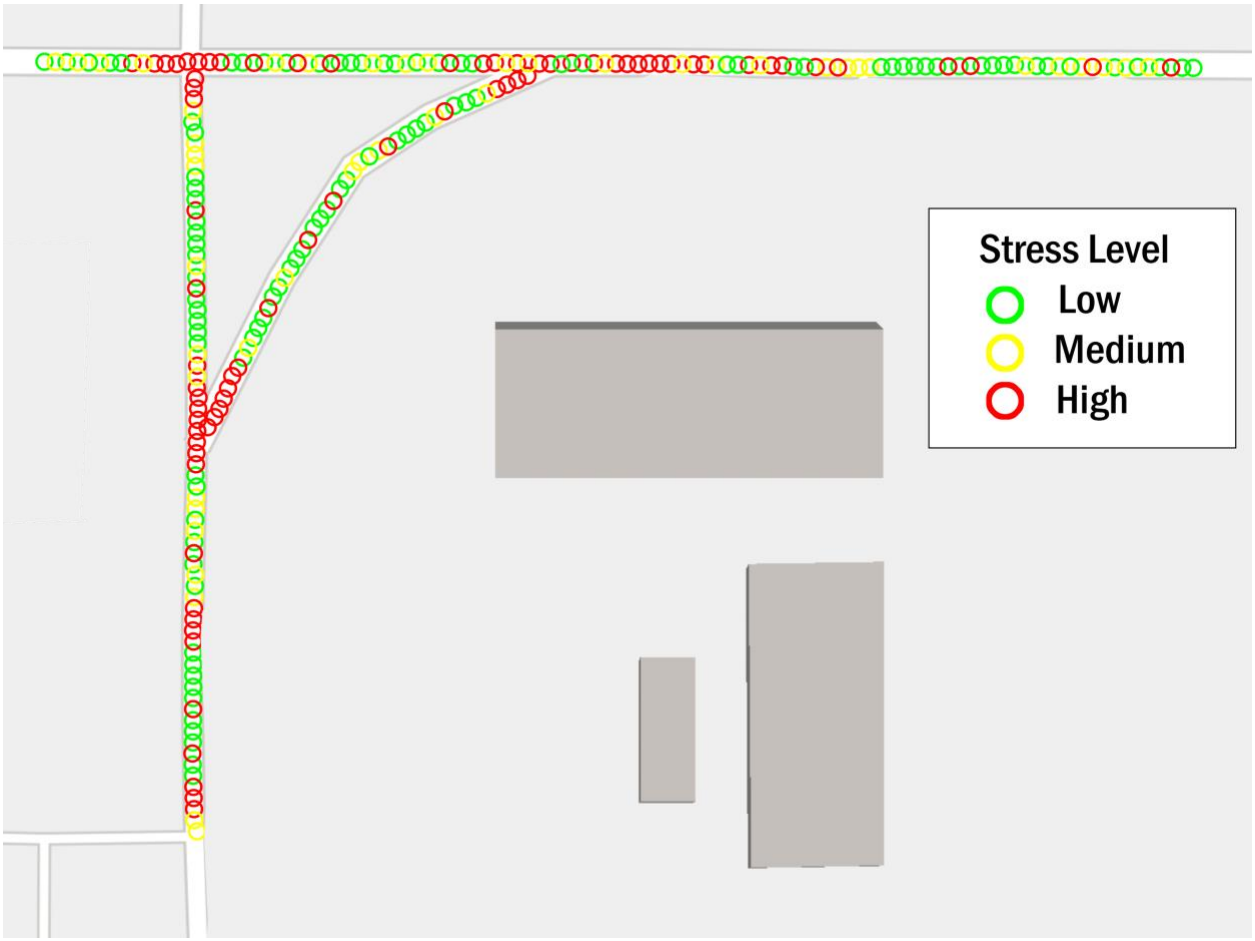


Figure 12: Cyclist Stress Distribution Heatmap

Chapter 6: EVALUATION

6.1 Experimental Setup

For system development and conducting experiments, the following hardware and software were used:

- Smartphone Application:

  OS: Android [46]

  Programming Language: Kotlin [47]

  Device specifications: Samsung Galaxy S10 smartphone with Octa-core Snapdragon 855 CPU, 8GB of RAM and inbuilt IMU sensors and microphone.

- Deep Learning Model:

  OS: Linux (Ubuntu 20.04)

  Programming Language: Python

  Deep learning framework: PyTorch [52] with CUDA acceleration

  GPU: NVIDIA Tesla V100-PCIE with 16GB RAM

  Supporting libraries: Numpy, Scikit-learn, Matplotlib, noisereduce, SciPy.

  Data Storage: Microsoft Azure cloud

- Deploying DL model on Android:

  Framework: Open Neural Network Exchange (ONNX) [58]

6.2 Data Collection and Field Testing

For the dataset collection for model training, a two-phase data collection campaign is executed. In the simulation phase, ten subjects participate in virtual bike riding across various scenarios, recording real-time stress levels at 5-second intervals. These scenarios include various road types such as urban, rural, and residential areas, different weather conditions like sunny, rainy, and foggy days, and varying traffic conditions with low, intermediate, and high volumes, along with different traffic light situations. Additionally, the presence or absence of bike lanes is also considered. Environmental cues are captured by an Android app through the smartphone's IMU sensor and microphone. Safety measures involve dividing data collection into two 30-minute sessions per participant with breaks and health monitoring. Participants also complete the simulator sickness questionnaire [53] during each session.

In the real-world phase, twenty subjects utilize the smartphone app during a 30-minute bike ride, resulting in a 10-hour real-world riding dataset. Unlike the simulation phase, stress levels are not reported in the real-world setting, facilitating convenient data collection during daily activities.

Figure 13: Smartphone Setup on Bike for Data Collection

To evaluate CyclistAI's real-world performance, field testing is conducted with twenty participants engaging in on-road biking sessions along their usual routes. They provide real-time stress level reports used as labels. To minimize distractions, participants do not manually input or visually shift their focus. Stress labels are recorded by a separate phone mounted on the bike's handlebars, extracted from audio and IMU sensor recordings.

6.3 Evaluation metrics

To thoroughly assess CyclistAI, a range of evaluation metrics is employed. These include accuracy, which measures the ratio of correct predictions to total samples, offering an overview of model effectiveness. Root Mean Square Error (RMSE) calculates the average prediction error, with a focus on larger errors. The F1 score considers both positive and negative samples, particularly in multi-class classification tasks where negative samples represent instances not belonging to the class, using a one-vs-rest strategy. The Receiver Operating Characteristic (ROC) curve plots the true positive rate against the false positive rate, providing insights into the model's performance across different thresholds. The Area Under the ROC Curve (AUC) quantifies the overall performance of the model in distinguishing between classes. This comprehensive set of metrics aids in evaluating CyclistAI across various aspects of its predictive capability and robustness.

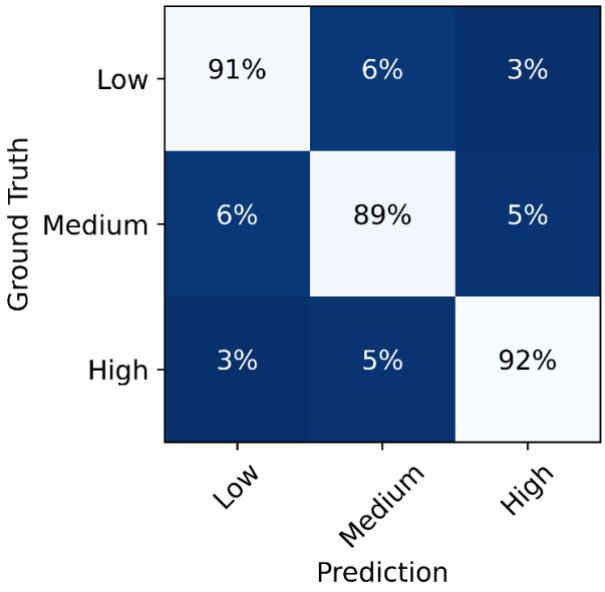


Figure 14: Confusion Matrix of Stress Levels

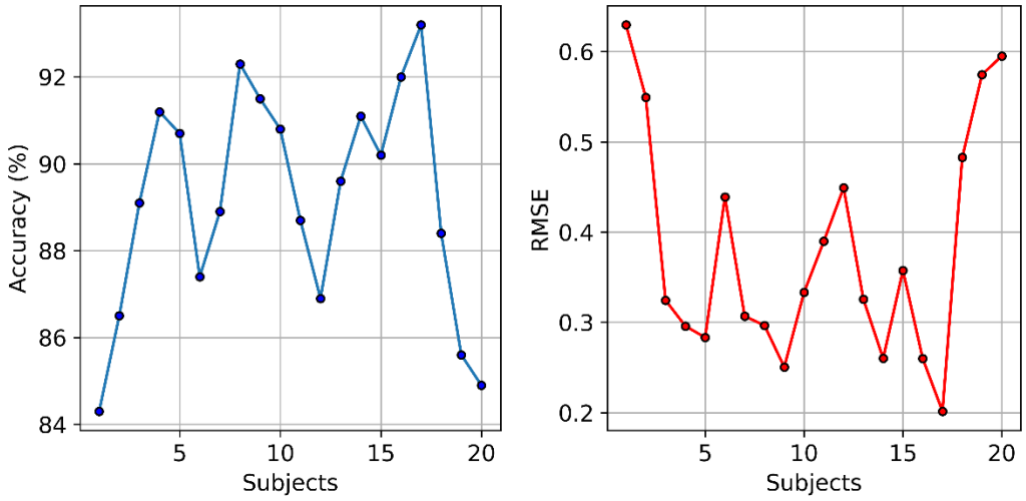
6.4 System Performance

Figure 14 illustrates the confusion matrix showing the correspondence between ground truth and predicted results. Each row corresponds to a ground truth class: low, medium, and high stress levels, respectively, while each column represents a predicted class. The diagonal line indicates the success rate for predicting each class. CyclistAI demonstrates a promising success

rate ranging from 89% to 92%. Additionally, similar prediction performance is observed across all classes, highlighting the stability of CyclistAI.

Next, evaluating the robustness of CyclistAI under cross-domain settings involves investigating its performance across subjects and various environments.

6.4.1 Subject-based Robustness Analysis



(a) Accuracy (b) RMSE

Figure 15: Subject-based Robustness Analysis

Subjects exhibit a wide range of understanding and interpretation when it comes to stress. This prompts an intriguing inquiry into the applicability of CyclistAI across diverse user profiles. To delve into this matter, CyclistAI is put to the test across twenty subjects in real-world scenarios. Figure 15 provides a visual representation of the accuracy and RMSE on a per-subject basis. The accuracy spans from 84.3% to 93.2%, showcasing consistency across a spectrum of subjects, with a standard deviation of 2.9%. However, a closer look reveals a relatively higher RMSE, suggesting a tendency for CyclistAI to generate predictions with larger errors for specific individuals. Despite this, the overall accuracy remains robust, attributed to the correlation between environmental cues and a cyclist's stress levels. This underscores the adaptability of CyclistAI across varied user profiles.

6.4.2 Environmental Robustness Analysis

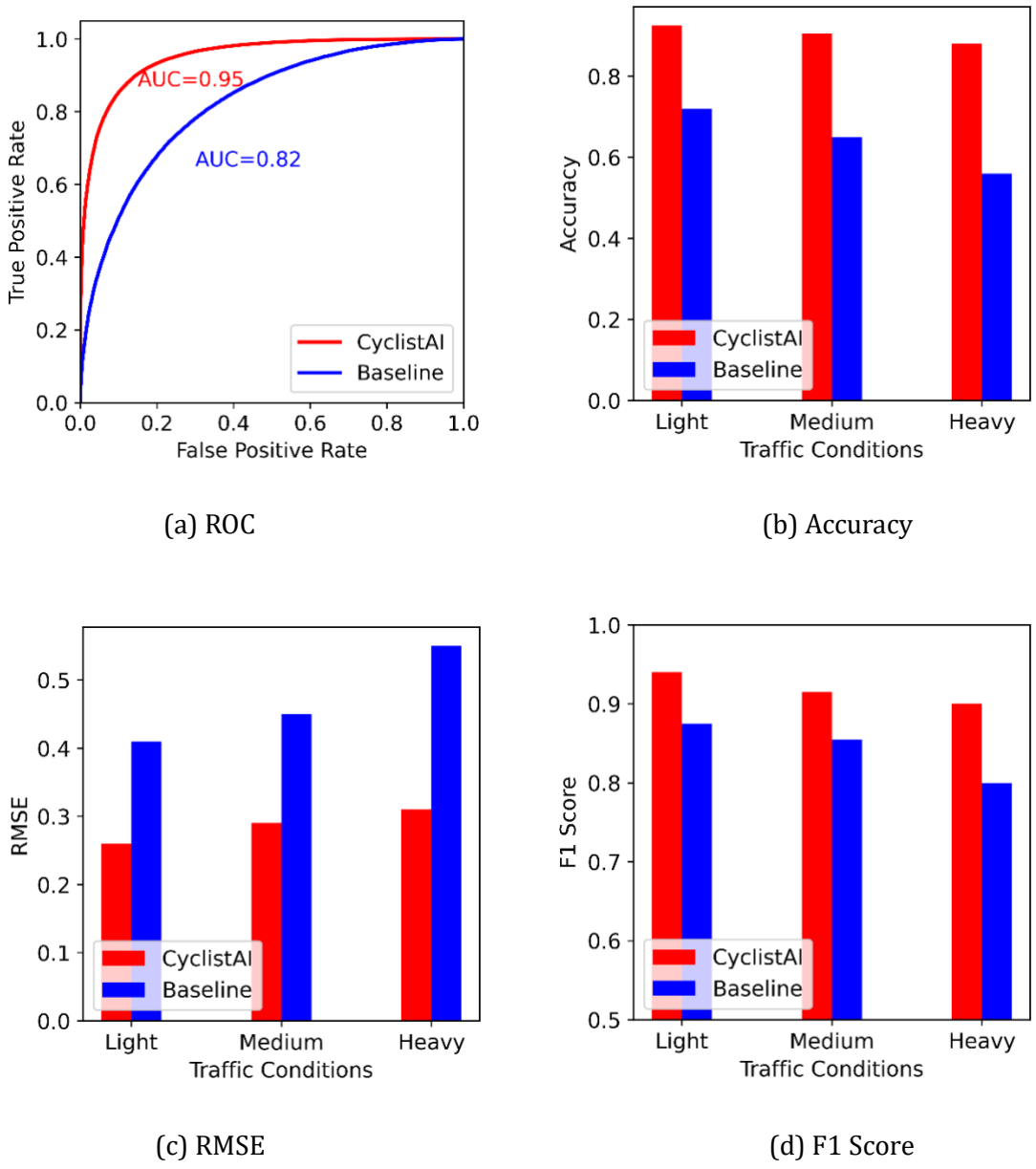
Environmental conditions significantly influence how cyclists perceive stress. Considering this, in-field testing examines three variable factors: time of day, weather conditions, and type of location. Each factor is studied independently while keeping the others constant. Table 1 showcases CyclistAI's performance across various environmental settings. The most challenging scenario occurs on urban roads, characterized by the presence of numerous objects such as pedestrians, traffic signals, vehicles, and road signs. However,
CyclistAI consistently maintains high accuracy, surpassing 84% across all settings. This sustained high performance underscores CyclistAI's effectiveness, especially when trained with cross-domain contrastive learning aided by a bike simulator, enabling it to seamlessly integrate information from diverse real-world environments post-training.

Table 1: Environmental Robustness Analysis

Metric	Time of Day		Weather Condition			Location Type		
	Day	Night	Sunny	Cloudy	Rainy	Campus	Residential	Urban
Accuracy	0.92	0.87	0.90	0.89	0.87	0.88	0.93	0.84
RMSE	0.30	0.54	0.26	0.29	0.54	0.32	0.26	0.44
F1 Score	0.91	0.91	0.94	0.93	0.90	0.88	0.91	0.87
AUC	0.97	0.89	0.93	0.96	0.92	0.92	0.94	0.90

6.5 Comparison with Prior Approach

To illustrate the superiority of CyclistAI, a comparison was conducted with a state-of-the-art solution utilizing bio signals such as heart rate (HR) [57] and respiration rate (RR) [56] for predicting cyclist stress. Employing a support vector machine [55] classifier trained on a dataset incorporating HR and RR measurements collected during in-field testing, ROC curves were generated for both CyclistAI and the baseline model. The ROC curve of CyclistAI demonstrates a higher true positive rate with a steeper shape, suggesting superior performance. Additionally, CyclistAI achieves a relatively larger area under the curve (AUC) of 0.95 compared to the baseline's 0.82, as shown in Figure 16a, indicating a notable performance advantage.



(a) ROC

(b) Accuracy

(c) RMSE

(d) F1 Score

Figure 16: Comparison of CyclistAI with Baseline

To further assess the effectiveness of CyclistAI, an examination of its performance metrics - accuracy, RMSE, and F1 score - across different traffic conditions was conducted. This analysis is presented in Figures 16b, 16c, and 16d. The following observations were made:

1. Consistent outperformance of the baseline method was observed by CyclistAI, showing higher accuracy and F1 scores while maintaining lower RMSE values across all traffic conditions.

2. Regardless of traffic volumes, stable prediction performance was demonstrated by CyclistAI, with a narrow standard deviation in accuracy (ranging from 2.5% to 3.9%). In contrast, more variability in performance metrics across different traffic scenarios was exhibited by the baseline method.

3. Particularly in heavy traffic situations, significant superiority over the baseline method was shown by CyclistAI, with a 32% increase in accuracy, along with reductions of 0.22 in RMSE and 0.10 in F1 score. This superiority can be attributed to the effective interpretation of subtle environmental cues associated with the stress levels by CyclistAI, which proved challenging for the baseline method.

Chapter 7: CONCLUSION AND FUTURE WORK

This project introduces CyclistAI, a pioneering model aimed at efficiently assessing cyclist stress by leveraging smartphone sensors. Integrating Convolutional Neural Networks and Long Short-Term Memory networks, CyclistAI translates environmental stimuli into quantifiable stress levels. The dataset utilized for training encompasses various traffic scenarios, including data from both a meticulously designed bike simulator and real-world observations. To bridge the simulation-to-reality gap, this study incorporates domain adaptation technique alongside a robust contrastive learning framework. By leveraging a combination of simulated and real-world data for training, CyclistAI is optimized to perform effectively in real-world scenarios, enhancing its overall performance and applicability.

Through extensive experimentation and thorough baseline comparisons, CyclistAI demonstrates promising capabilities in accurately gauging cyclist stress levels. Furthermore, this project sheds light on the implications of CyclistAI in enhancing cyclist safety and urban planning strategies by providing actionable insights derived from real-time stress assessments.

For future work, various possibilities present themselves. Firstly, enhancing the energy efficiency of the data acquisition process remains a crucial area of exploration. Implementing advanced techniques to optimize smartphone power consumption while maintaining accuracy will be pivotal for the widespread deployment of CyclistAI in real-world settings. Additionally, further investigation into the integration of additional sensor modalities, such as GPS or heart rate monitors, could provide richer data inputs for more comprehensive stress assessment.

In conclusion, this project lays the groundwork for a novel approach to cyclist stress assessment with CyclistAI. By addressing key challenges and showcasing promising results, it sets the stage for future research endeavors aimed at advancing the field of humancentric computing and urban mobility solutions.

REFERENCES

[1] TA Litman. 2010. Short and Sweet: Analysis of Shorter Trips Using National Personal Travel Survey. Victoria Transport Police Institute (2010).

[2] AASHTO Executive Committee et al. 1999. Guide for the development of bicycle facilities. American Association of State Highway Transportation Officials: Washington, DC, USA (1999).

[3] Søren Underlien Jensen. 2007. Pedestrian and bicyclist level of service on roadway segments. Transportation research record 2031, 1 (2007), 43-51.

[4] Bryan Blanc and Miguel Figliozzi. 2016. Modeling the impacts of facility type, trip characteristics, and trip stressors on cyclists' comfort levels utilizing crowdsourced data. Transportation Research Record 2587, 1 (2016), 100-108.

[5] Greg Rybarczyk, Ayse Ozbil, Erik Andresen, and Zachary Hayes. 2020. Physiological responses to urban design during bicycling: A naturalistic investigation. Transportation research part F: traffic psychology and behaviour 68 (2020), 79-93.

[6] Lu´ıs Rita, Miguel Peliteiro, Tudor-Codrin Bostan, Tiago Tamagusko, and Adelino Ferreira. 2023. Using Deep Learning and Google Street View Imagery to Assess and Improve Cyclist Safety in London. Sustainability 15, 13 (2023), 10270.

[7] April Gadsby, James Tsai, and Kari Watkins. 2022. Understanding the influence of pavement conditions on cyclists' perception of safety and comfort using surveys and eye tracking. Transportation research record 2676, 12 (2022), 112-126.

[8] April Gadsby, Marjan Hagenzieker, and Kari Watkins. 2021. An international comparison of the self-reported causes of cyclist stress using quasi-naturalistic cycling. Journal of Transport Geography 91 (2021), 102932.

[9] Alex Sorton and Thomas Walsh. 1994. Bicycle stress level as a tool to evaluate urban and suburban bicycle compatibility. Transportation Research Record (1994).

[10] Jorge A Huertas, Alejandro Palacio, Marcelo Botero, Germa´n Carvajal, Thomas van Laake, Diana Higuera-Mendieta, Sergio A Cabrales, Luis A Guzman, Olga L Sarmiento, and Andr´es L Medaglia. 2020. Level of traffic stress-based classification: a clustering approach for Bogot´a, Colombia. Transportation Research Part D: Transport and Environment 85 (2020), 102420.

[11] Shumayla Yaqoob, Salvatore Cafiso, Giacomo Morabito, and Giuseppina Pappalardo. 2023. Deep transfer learning-based anomaly detection for cycling safety. Journal of Safety Research (2023).

[12] Dayan Guan, Jiaxing Huang, Shijian Lu, and Aoran Xiao. 2021. Scale variance minimization for unsupervised domain adaptation in image segmentation. Pattern Recognition 112 (2021), 107764.

[13] Yanchao Yang and Stefano Soatto. 2020. Fda: Fourier domain adaptation for semantic segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 4085-4095.

[14] Samuel Schmidgall, Jascha Achterberg, Thomas Miconi, Louis Kirsch, Rojin Ziaei, S. Pardis Hajiseyedrazi, and Jason Eshraghian. 2023. Brain-inspired learning in artificial neural networks: a review. arXiv:2305.11252 (2023).

[15] John Douglas Hunt and John E Abraham. 2007. Influences on bicycle use. Transportation 34 (2007), 453-470.

[16] Rachel L Vickery. 2008. The effect of breathing pattern retraining on performance in competitive cyclists. Ph.D. Dissertation. Auckland University of Technology.

[17] Xiaomin Ouyang, Xian Shuai, Jiayu Zhou, Ivy Wang Shi, Zhiyuan Xie, Guoliang Xing, and Jianwei Huang. 2022. Cosmo: Contrastive Fusion Learning with Small Data for Multimodal Human Activity Recognition. Proceedings of the 28th ACM Conference on Mobile Computing And Networking. 324-337.

[18] Ronan Doorley, Vikram Pakrashi, Eoin Byrne, Samuel Comerford, Bidisha Ghosh, and John A Groeger. 2015. Analysis of heart rate variability amongst cyclists under perceived variations of risk exposure. Transportation research part F: traffic psychology and behaviour 28 (2015), 40-54.

[19] Shiv Ram Dubey, Satish Kumar Singh, and Bidyut Baran Chaudhuri. 2019. Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark. arXiv:2109.14545 (2019).

[20] Dillon T Fitch, James Sharpnack, and Susan L Handy. 2020. Psychological stress of bicycling with traffic: examining heart rate variability of bicyclists in natural urban environments. Transportation research part F: traffic psychology and behaviour 70 (2020), 81-97.

[21] T Jones, K Chatterjee, J Spinney, E Street, C Van Reekum, B Spencer, H Jones, LA Leyland, C Mann, S Williams, et al. 2016. Cycle BOOM: Design for lifelong health and wellbeing. In Summary of key findings and recommendations. Oxford Brookes University UK.

[22] Sylvain Laborde, Emma Mosley, and Julian F Thayer. 2017. Heart rate variability and cardiac vagal tone in psychophysiological research recommendations for experiment planning, data analysis, and data reporting. Frontiers in psychology 8 (2017), 213.

[23] Sebastian Ruder. 2017. An overview of gradient descent optimization algorithms. arXiv:1609.04747v2 (2017).

[24] 2021. US Smartwatch Revenue To Grow By Almost 20% And Pass $10B Mark-$10.4B in 2021.https://tradingplatforms.com/blog/2021/07/08/us-smartwatchrevenue-to-grow-y-almost-20-and-pass-10bmark-10-4b-in-2021.

[25] Taehee Lee, Chanjun Chun, and Seung-Ki Ryu. 2021. Detection of road-surface anomalies using a smartphone camera and accelerometer. (2021), 561.

[26] Prashanth Mohan, Venkata N Padmanabhan, and Ramachandran Ramjee. 2008. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In Proceedings of the 6th ACM conference on Embedded network sensor systems. 323-336.

[27] Dinesh Vij and Naveen Aggarwal. 2018. Smartphone based traffic state detection using acoustic analysis and crowdsourcing. Applied Acoustics 138 (2018), 80-91.

[28] Munshi Yusuf Alam, Shahrukh Imam, Harshit Anurag, Sujoy Saha, Subrata Nandi, and Mousumi Saha. 2018. LiSense: Monitoring city street lighting during night using smartphone sensors. In 2018 IEEE international conference on data mining workshops (ICDMW). IEEE, 596-603.

[29] Shona L Halson, MatthewWBridge, Romain Meeusen, Bart Busschaert, Michael Gleeson, David A Jones, and Asker E Jeukendrup. 2002. Time course of performance changes and fatigue markers during intensified training in trained cyclists. Journal of applied physiology 93, 3 (2002), 947-956.

[30] Xiaofeng Liu, Chaehwa Yoo, Fangxu Xing, Hyejin Oh, Georges El Fakhri, Je-Won Kang, and Jonghye Woo. 2022. Deep Unsupervised Domain Adaptation: A Review of Recent Advances and Perspectives. arXiv:2208.0742 (2022).

[31] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. Deep Learning. Cambridge, MA: MIT Press.

[32] Sepp Hochreiter, and Ju¨rgen Schmidhuber. 1997. Long Short-Term Memory. Neural Computation 9, no. 8 (1997): 1735-1780.

[33] EA Bjo¨rk. 1999. Startle, annoyance and psychophysiological responses to repeated sound bursts. Acta Acustica united with Acustica 85, 4 (1999), 575-578.

[34] Howard S Hoffman and John L Searle. 1968. Acoustic and temporal factors in the evocation of startle. The Journal of the Acoustical Society of America 43, 2 (1968), 269-282.

[35] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradientbased learning applied to document recognition. Proceedings of the IEEE 86, no. 11 (1998): 2278-2324.

[36] Serkan Kiranyaz, Onur Avci, Osama Abdeljaber, Turker Ince, Moncef Gabbouj, and Daniel J. Inman. 2019. 1D Convolutional Neural Networks and Applications: A Survey. arXiv:1905.03554 (2019).

[37] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In International conference on machine learning. PMLR, 1597-1607.

[38] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. 2020. What makes for good views for contrastive learning? Advances in neural information processing systems 33 (2020), 6827-6839.

[39] Chen Chen, Jason C Anderson, Haizhong Wang, Yinhai Wang, Rachel Vogt, and Salvador Hernandez. 2017. How bicycle level of traffic stress correlate with reported cyclist accidents injury severities: A geospatial and mixed logit analysis. Accident Analysis & Prevention 108 2017), 234-244.

[40] Nicholas N Ferenchak and Wesley E Marshall. 2020. Validation of bicycle level of traffic stress and perceived safety for children. Transportation research record 2674, 4 (2020), 397-406.

[41] James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Vol. 1. Oakland, CA, USA, 281-297.

[42] Andrew J Filtness, Andrew Fox, and John Phillips. 2016. Cognitive Workload During Cycling: Sensitivity of Physiological and Subjective Measures. Ergonomics 59, no. 4 (2016). 563-572.

[43] Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R. Arabnia. 2020. A Brief Review of Domain Adaptation. arXiv:2010.03978 (2020).

[44] Matthew Wilson, Gavin Wood, and Brendan Elliott. 2009. Anxiety, stress, and coping mechanisms in competitive sport. In T. S. Horn (Ed.), Advances in sport psychology (3rd ed. 202-221).

[45] Zhilu Zhang, and Mert R. Sabuncu. 2018. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. arXiv:1805.07836 (2018).

[46] Android Operating System. 2008. https://www.android.com/

[47] Kotlin Language. 2011. https://kotlinlang.org/

[48] Python. 1991. https://www.python.org/

[49] Jiaxing Huang, Dayan Guan, Aoran Xiao, Shijian Lu, and Ling Shao. 2022. Category contrast for unsupervised domain adaptation in visual tasks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 1203-1214.

[50] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. 2019. Bidirectional learning for domain adaptation of semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 6936-6945.

[51] J Semotan, M Semotanova, and M Oldwa. 1969. Startle and other human responses to noise. Journal of Sound and Vibration 10, 3 (1969), 480-488.

[52] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, et al. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. arXiv:1912.01703 (2019).

[53] Robert S Kennedy, Norman E Lane, Kevin S Berbaum, and Michael G Lilienthal. 1993. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. The international journal of aviation psychology 3, 3 (1993), 203-220.

[54] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised Contrastive Learning. arXiv:2004.11362(2020).

[55] Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. AT&T Bell Labs (1995).

[56] Joachim Taelman, Steven Vandeput, Arthur Spaepen, and Sabine Van Huffel. 2009. Influence of mental stress on heart rate and heart rate variability. In 4th European Conference of the International Federation for Medical and Biological Engineering: ECIFMBE 2008 23-27 November 2008 Antwerp, Belgium. Springer, 1366-1369.

[57] Paul Grossman. 1983. Respiration, stress, and cardiovascular function. Psychophysiology 20, 3 (1983), 284-300.

[58] Open Neural Network Exchange (ONNX). 2017. https://onnx.ai/

[59] Jongseo Sohn, Nam Soo Kim, and Wonyong Sung. 1999. A statistical model-based voice activity detection. IEEE signal processing letters 6, 1 (1999), 1-3.

[60] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2012. On the difficulty of training Recurrent Neural Networks. arXiv:1211.5063 (2012).

Appendix A:

Appendix B: Technology Transfer

An Appendix should be included in this final report to document the Technology Transfer activities conducted during the project term, accomplishments towards T2 adoption and implementation by relevant stakeholders, as well as any relevant post-project T2 plans.

CENTER FOR TRANSPORTATION EQUITY, DECISIONS & DOLLARS

The **Center for Transportation, Equity, Decisions and Dollars (CTEDD)** is a USDOT University Transportation Center, leading **transportation policy research** that aids in **decision making** and **improves economic development** through **more efficient, and cost-effective use of existing transportation systems,** and offers **better access to jobs and opportunities.** We are leading a larger consortium of universities focused on **providing outreach** and **research to policy makers**, through **innovative methods** and **educating future leaders of the transportation field.**

UNIVERSITY OF TEXAS ARLINGTON

CAL POLY SAN LUIS OBISPO

Georgia Tech

USF UNIVERSITY OF SOUTH FLORIDA

WISCONSIN UNIVERSITY OF WISCONSIN–MADISON

C-TEDD@UTA.EDU CTEDD.UTA.EDU @UTACTEDD @ C__TEDD @ CENTER FOR TRANSPORTATION, EQUITY, DECISIONS AND DOLLARS 817.272.5138

The project "Assessing Cyclist's Stress on A Large-Scale: A Practical Smartphone-Based Data-Driven Approach" presents a highly innovative and practical solution for cyclist stress assessment, combining smartphone sensing and advanced deep learning models. By relying on widely accessible smartphone sensors, such as the IMU and microphone, the project introduces a cost-effective and accessible approach that doesn't require specialized equipment, making it highly practical for large-scale deployment.

The project also showcases its broader potential impact through its proposal to aggregate individual stress assessments into visualized stress distribution maps. This has significant implications for urban planning and infrastructure development, as city planners could use these maps to identify high-stress areas and prioritize improvements, such as adding bike lanes or reducing traffic congestion. The research outcome could contribute to creating safer, more cyclist-friendly cities, thus encouraging more people to adopt cycling as a mode of transportation, which in turn promotes sustainability, public health, and environmental benefits.

Overall, this project represents a pioneering effort with far-reaching potential to not only improve cyclist safety and well-being but also support cities in developing more sustainable transportation systems. The project's approach could easily be adapted to other forms of transportation or stress-related assessments in various urban contexts.

Reviewer

Mingyan Xiao, PhD.
Assistant Professor
Computer Science, College of Science
CalPolyPomona
Contact: mxiao@cpp.edu

Overall Evaluation:

The project offers a strong technical contribution to the field of cyclist stress assessment by developing a scalable, real-time system that leverages deep learning and smartphone sensors. The robustness of the system across different environments and the use of domain adaptation techniques are key strengths that set this work apart. In the meantime, this project may consider enhancing privacy and discussing long-term deployment considerations to fully realize the system's potential in real-world applications.

Strengths:

The development of CyclistAI presents a highly innovative solution by leveraging widely available smartphone sensors to assess cyclist stress. This approach is practical, scalable, and offers a cost-effective alternative to more expensive physiological measures, which are often limited in large-scale deployments.

The model's ability to provide real-time stress assessments is a major advantage over traditional methods that rely on retrospective data (such as self-report surveys). This improvement addresses recall biases and allows for a more accurate capture of cyclists' immediate experiences.

The combination of CNN and LSTM for feature extraction and temporal analysis reflects a well-considered approach to processing sensor data. The report offers clear technical justifications for using these architectures, highlighting their strengths in detecting patterns in both IMU and audio data.

The report provides a thorough evaluation of CyclistAI across different traffic environments, weather conditions, and time of day. The analysis is supported by key metrics such as accuracy, F1 score, and RMSE, which helps validate the robustness of the model in real-world scenarios.

Suggestions for improvements:

Discuss how CyclistAI can safeguard the privacy of users, especially regarding audio and location data. Including privacy-preserving techniques such as differential privacy or on-device encryption would help alleviate potential user concerns.

Include considerations for energy-efficient data collection and long-term usability of the application. This would ensure that the system remains practical for daily use without significantly draining smartphone batteries.

Reviewer

Linke Guo Ph.D., SMIEEE
Associate Professor
Department of Electrical and Computer Engineering
Clemson University
Contact: linkeg@clemson.edu

The Department of Computer Science
University Plaza, Atlanta Georgia 30303
404-413-5717 (Fax)
wli28@gsu.edu

Sep. 12, 2024

Re: Review request for CTEDD-supported project "**Assessing Cyclist's Stress on A Large-Scale: A Practical Smartphone-Based Data-Driven Approach**"

The CTEDD-supported project titled "**Assessing Cyclist's Stress on A Large-Scale: A Practical Smartphone-Based Data-Driven Approach**" develops CyclistAI, a novel system to assess cyclist stress levels using smartphone sensors (IMU and microphone) combined with deep learning techniques. The aim is to create a scalable, real-time assessment model that leverages Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to analyze contextual data from the cycling environment, including traffic and road conditions. The system addresses existing limitations in cyclist stress assessments, such as the impracticality of current frameworks and the challenges in real-time data collection. The report also highlights the use of domain adaptation to overcome the gap between simulation and real-world data for training, using contrastive learning techniques. CyclistAI is tested on both simulation and real-world datasets, achieving high accuracy and demonstrating robustness across different traffic and environmental conditions. The idea of aggregating stress assessments into a visualized stress map is both innovative and practical. This has clear implications for urban planners, allowing them to use real-time data to improve cycling infrastructure and promote safer cycling environments.

As a potential future direction, the researcher team may consider exploring integration with wearable technologies to provide more comprehensive insights into cyclist stress.

Weili

Wei (Lisa) Li, Ph.D.
Associate Professor
Associate Director of Graduate Studies
Department of Computer Science
Georgia State University