UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

Temporal Intelligence in Wearable Health: Towards Robust and Reliable Gesture-Based AI Frameworks

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

BY

Hifza Afzal

Tulsa, Oklahoma

2026

Temporal Intelligence in Wearable Health: Towards Robust and Reliable Gesture-Based AI Frameworks

A DISSERTATION APPROVED FOR THE

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. Chad Roller, Chair

Dr. James J. Sluss, Jr.

Dr. Samuel Cheng

Dr. Timothy Ford

Abstract

The widespread adoption of smartwatches and wrist-worn wearables presents a unique opportunity to enable continuous, privacy-preserving health monitoring in everyday life. This dissertation develops a series of AI-driven frameworks that leverage inertial and multimodal sensor data from commodity smartwatches to monitor clinically significant health behaviors. Rather than relying on intrusive audio or video recordings, the proposed systems interpret characteristic hand and arm gestures associated with the health events, enabling the device to act intelligently and selectively without compromising user privacy. Across all the frameworks, the work progressively advances from lightweight models to compact deep learning architectures and transformer-based models with modern positional encoding strategies, each tailored to the unique temporal and motion characteristics of the target behavior. Collectively, this work establishes that temporal intelligence embedded in wrist-worn devices can serve as a practical, scalable, and privacy-respecting foundation for longitudinal health monitoring, with meaningful implications for early intervention, medication adherence, and clinical decision support.

Acknowledgments

I would like to express my sincere appreciation to Dr. Chad Roller for his guidance and support during the final stages of this dissertation. His insights, encouragement and willingness to assist in bringing this work to completion are greatly appreciated.

I would also like to thank the members of my dissertation committee Dr. James J. Sluss, Dr. Timothy G. Ford, and Dr. Samuel Cheng for their time, valuable feedback and thoughtful contributions. I also extend my sincere appreciation to Denise Davis, Prof. Hjalti Sigmarsson, Krista Pettersen and Hannah James for their exceptional administrative support during my studies at OU-Tulsa.

I am deeply grateful to Dr. Ali Imran whose exceptional guidance and mentorship were instrumental in the completion of this dissertation. I will always remain indebted to Dr. Imran for instilling in me the qualities of a diligent researcher and for providing invaluable advice on excelling in both personal and professional aspects of life.

I am profoundly grateful to my family for their unwavering support throughout this challenging journey. I owe a deep debt of gratitude to my parents, Ami and Baba, whose prayers, countless sacrifices and constant encouragement have been the foundation of my success. I am equally thankful to my siblings Sumia, Mona, Asad and Puchki for their continuous support, encouragement and belief in me throughout this journey.

I extend my heartfelt thanks to my beloved husband, Shahrukh, for his unwavering belief in me even during the moments when I struggled to believe in myself. His constant support and mentorship throughout every phase of this journey have been a profound source of strength and inspiration.

Finally, I express my sincere gratitude to my manager Eric and my colleagues at DATUM. Their encouragement, collaboration and support have been invaluable throughout my academic journey.

Table of Contents

List of Figures

List of Tables

CHAPTER 1

Coughalytics: An "About to Cough" Spotting through Smartwatch-Enabled Gesture Detection

1.1 Introduction

1.1.1 Motivation

Cough, is often associated with various respiratory diseases affecting a significant global population. The most prevalent chronic respiratory diseases such as Chronic Obstructive Pulmonary Disease (COPD) and asthma contribute to nearly 4 million deaths each year, as reported by the World Health Organization [1]. Timely prediction and prompt treatments for these respiratory diseases are vital in preventing major causes of death.

Recent studies have demonstrated that continuous monitoring of cough frequency is the most effective predictor for exacerbation prediction in COPD and asthma [2]. Clinical studies, such as

those in [3], demonstrate that a simple always-on domiciliary audio recording-based cough counting tool can effectively predict and prevent acute exacerbation of COPD. It can provide an advance warning of 3.4 days, with a significantly lower false alarm rate compared to widely used tools like the Leicester Cough Questionnaire (LCQ) or other daily questionnaires [4].

The clinical significance of monitoring and analyzing patient-specific cough has led to regulatory approvals and commercialization of several cough monitoring tools [5, 6]. These tools typically include an always-on microphone attached to the chest or neck, along with a portable recording device [7, 8]. During the monitoring period, proprietary algorithms or trained individuals extract the cough count from the recorded ambulatory sounds. Several studies, including experimental and clinical trials, have showcased the potential of these tools. Nonetheless, current cough monitoring devices lack the necessary features of privacy protection and suitable form factors for prolonged use and easy wearability.

In contrast, smartwatches have gained significant traction and are now a staple worn by people throughout their daily routines. Inspired by this unmet need this chapter introduces the development of the world's first "About to Cough" (AC) gesture spotting via smartwatch. By leveraging the Inertial Measurement Unit (IMU) data stream of the watch, it is feasible to create a lightweight algorithm that automatically activates/deactivates the microphone only when a user is "About to cough" (AC) and stops recording when the cough finishes, through a simple few-shot learning method. This novel solution will not only provide privacy protection, but also storage when compared to the state-of-the-art always on cough recording methods, enabling for the first time the desired longitudinal use, a critical advancement in this field.

1.1.2 Contributions and Organization

In essence, this report outlines the following significant contributions:

- We propose a novel mechanism that not only predicts coughs from smartwatches but also

detects the AC gesture before cough is fully completed in real-time and continuous manner. By activating the smartwatch's microphone in anticipation of the cough sound, our approach ensures accurate and uninterrupted capture of the entire cough event without any post-completion delays.

- To safeguard privacy and maintain confidentiality, the microphone is exclusively activated during a cough event, ensuring that it does not capture any other audio content.
- The key strength of the proposed approach lies in its few-shot learning/personalized capabilities to provide accurate and personalized detection of cough gestures.
- Data collection and detection are seamlessly handled through our customized Android application, providing a user-friendly experience while ensuring data accuracy and reliability.
- We developed a framework using a lightweight AC gesture spotting algorithm utilizing warping longest common subsequence (wLCSS) in our framework. This advanced algorithm eliminates the necessity for users to explicitly mark gesture boundaries, ensuring a smooth and intuitive experience with constant or linear time and space complexity. This optimization will allow the algorithm to make timely inferences without overburdening the watch's memory and processor. As a result, the smartwatch can seamlessly run other tasks alongside AC gesture spotting without compromising performance.
- Our research methodology emphasized developing an AC gesture spotting algorithm to account for varying individuals in both the AC gestures, ensuring accurate detection and recognition of cough gestures across different individuals.

This chapter is organized into four main sections. Section 1.2 outlines the challenges of AC gesture spotting, emphasizing the unique constraints not commonly addressed in existing gesture recognition algorithms. Section 1.3 details the proposed framework, from data collection and combination to personalized gesture template library creation and real-time detection. Section 1.4 and 1.5 presents a comprehensive study involving gesture analysis, participant testing, and an

in-depth examination of the framework's performance, including execution times and accuracy metrics. The report concludes by summarizing the achieved recognition performance of AC from the experiments, showcasing the promising usability and future prospects of the proposed framework.

1.2 State of the art

AC gesture spotting presents a more challenging research problem. This is due to its need to satisfy all constraints discussed above, which are not commonly addressed together in most state-of-the-art gesture spotting/recognition algorithms found in the existing literature [7–14, 14, 15]. The question at hand is: *Why current gesture spotting solutions are unsuitable for AC?*

In recent years, many academic studies and commercial solutions have demonstrated the potential of IMU data for identifying dynamic hand and arm gestures [9]. Most of these studies, however, are focused on offline gesture spotting i.e., through retro processing of the collected IMU data after gesture completion, and not spotting a gesture as it is happening. Some proposed schemes do enable online gesture spotting e.g., through dynamic time warping (DTW) [10] and Hidden Markov Models (HMM)-based methods [11]. These methods, however, are sensitive to IMU noise [12] and watch orientation [10]. The inability to cope with inter and intra-person variations is another challenge in existing gesture spotting solutions. To address these problems, more sophisticated training data-intensive approaches such as decision trees [13] and deep learning models [14] to differentiate gestures with fine granularity are proposed. For example, recently a very deep CNN-based approach implemented on a custom watch with a modified kernel to allow IMU sampling at 4000Hz is shown to detect high granularity hand gestures with 95.7% accuracy in the presence of inter and intra-person gesture variability [16]. While powerful for specific use cases where very fine gesture granularity (e.g., finger motion level) is needed necessitate a very high IMU sampling rate that is not viable for commodity watches. Also, these complex algorithms are shown to drain even an expensive watch battery within a few hours [15]. Therefore, despite their high accuracy, such

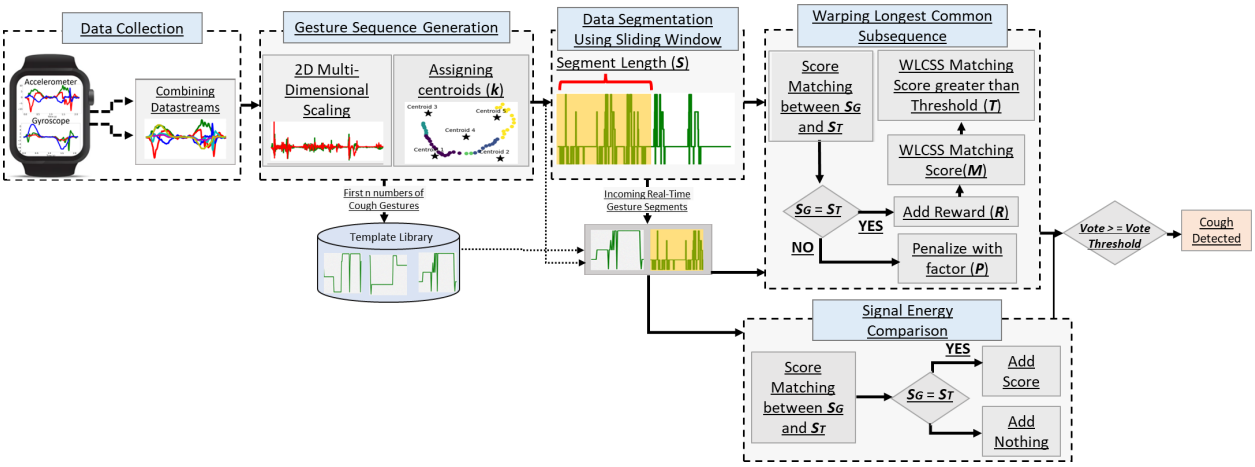

Figure 1.1: Overview of the proposed Coughlytics framework, illustrating the end-to-end pipeline from multimodal sensor data collection and preprocessing to sliding-window segmentation, personalized template generation, and wLCSS-based template matching for real-time cough detection.

approaches are not feasible.

1.3 Methodology

Our proposed framework aims to meet comprehensive requirements, including the early detection of coughs before completion and discreetly recording cough events through a few-shot learning method for AC gesture spotting. Our framework, depicted in Fig. 1.1, starting with a low complexity, yet highly flexible approach. We have structured our framework into several main components from data collection to template matching.

1.3.1 Data Collection

Our framework seamlessly gathers IMU data from smartwatches. This time series data is derived from continuous streams and encompasses readings from both the accelerometer and gyroscope along all three axes of the smartwatches. The data is sampled at a rate of 100Hz, a frequency that not only proves ample for the early detection of AC gestures but is also commonly supported by most commodity smartwatches.

We employed both accelerometer and gyroscope sensors in our study to address tilting/watch orientation issues. Though our proposed pre-processing offers robustness to both noise and some degree of tilting. However, severe tilting may deteriorate the performance if only accelerometer data is used. Unlike previous research [10], which solely relied on the 3D vectors of the accelerometer stream, our approach now utilizes 6D vectors that captures both linear and rotational movements of the smartwatches, as almost all smartwatches contain accelerometer and gyroscope sensors. This enhancement not only improves robustness against noise but also accounts for some degree of tilting. However, relying solely on accelerometer data can negatively impact performance.

1.3.2 Gesture Sequence Generation

The 6-dimensional feature vector of accelerometer and gyroscope now corresponds to one measurement sample which is further embedded to only two dimensions in the Euclidean space using Multidimensional Scaling (MDS) method primarily emphasizes preserving the relative distances between data points. If two data points are close in the 2D representation, it implies that they are similar or have similar patterns in the original 6D space.

Beyond its role in visualization, the transformation from 6D to 2D within our framework serves to enhance result interpretation and optimize computational efficiency. This efficiency is particularly crucial for achieving timely and reliable cough detection. Moreover, we optimize the clustering

process by initializing the k-means centroids *(k)* to align with the target clusters. Further, we employ the Euclidean distance calculation method by quantizing the data to efficiently assign data points to the nearest cluster. The clustering method serves two main objectives: creating a user gesture template and handling real-time incoming data streams discussed in upcoming sections.

1.3.3 Personalized Gesture Template Library

The basic etiquette of coughing is via hand gestures [17,18]. These gestures include swift movement of hand(s)/elbow towards covering the mouth, nose, or both. In our framework, we address three different etiquettes of coughing: the fist cough, elbow cough, and sleeve cough. The rationale behind selecting these three specific cough etiquettes was to ensure comprehensive coverage of the main coughing behaviors. Fig. 1.2 illustrates a range of gestures associated with three distinct etiquettes, alongside additional non-verbal cues. The highlighted gestures encompass not only coughing but also activities such as drinking water, yawning, and handling phone calls. The figure also serves as a visual representation of the 3D time series data, captured by accelerometer and gyroscope.

The user initially from the smartwatch collects *n* number of templates of cough gestures for template matching by using IMU data as a time series to store their specific AC gesture patterns. These templates are essential for template matching and serve to capture the unique AC gesture patterns associated with coughing. Subsequently, these templates are then cleaned and stored, with their respective means calculated, ensuring that only the cough event is retained while excluding any non-cough events from that template to enhance the accuracy of cough detection. The collected template data is then quantized and stored in a template library. These stored templates are then used to identify AC gestures in real-time by comparing the IMU stream against the stored templates using a similarity threshold.

1.3.4 Data Segmentation using Sliding Window

In real-time data streaming, we employ a sliding window technique to divide the data into segments *(S)*. This facilitates segmenting the data at regular intervals and moving along with the incoming data stream. This approach allows us to maintain our primary objective of promptly detecting coughs in real-time while keeping latency to a minimum.

Instead of utilizing an non-overlapping sliding window technique, we use a overlapping sliding window approach. This choice ensures that the detection of impending cough gestures remains swift, without introducing an additional delay.

To address a challenge in our cough detection system, we have identified the issue of counting single cough instances multiple times due to the cough pattern and sliding window. To mitigate this, we propose the incorporation of an additional feature called the "guard interval time". The guard interval time ensures that the system does not detect the same cough repeatedly during its completion. During this time, any potential cough signals will be disregarded to prevent detecting the remaining portion of the same cough. By incorporating this guard interval, the system avoids duplicate detections, reduces false positives, and improves overall detection accuracy while ensuring that each cough instance is counted only once.

1.3.5 Warping Longest Common Subsequence (wLCSS)

Our core framework relies on a template matching method, which operates on IMU data as a time series to create user-specific templates for AC gestures. These templates are then used to detect AC gestures by comparing the IMU real time data stream within the sliding window *(W)* to the stored user-specific AC templates, triggering a detection when similarity surpasses a predefined threshold. This approach is inspired by the Template Matching Methods (TMM) using DTW that has demonstrated very promising results for online gesture spotting with low computational

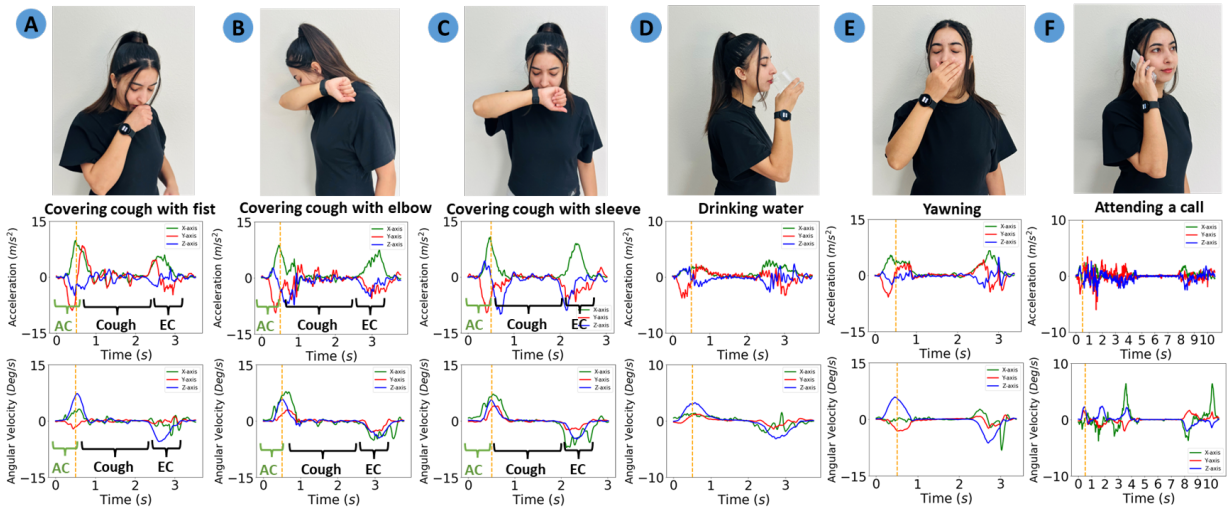

Figure 1.2: About-to-Cough (AC) and other common daily gestures, along with their corresponding 3-axis accelerometer and gyroscope IMU signals. The AC, cough, and end-of-cough (EC) phases are highlighted to illustrate temporal motion patterns across activities.

complexity [10]. Nevertheless, traditional DTW has demonstrated sensitivity to noise in time series data, a challenge inherent in smartwatches given their intended use in real-life scenarios.

As we addressed earlier, we propose a highly optimizable and personalized pre-processing scheme to tackle this issue. This method aims to convert the raw data from the IMU into adaptable-length segments *(S)* that will subsequently be transformed into centroids *(k)*. This pre-processing scheme not only achieves low complexity noise robustness but also enables faster inference, facilitating efficient spotting through a suitable TMM. By utilizing a similarity measure between the set of centroids *(k)*, a straightforward TMM method we propose to adapt the wLCSS as the TMM. This selection is based on two compelling reasons: 1) wLCSS offers significant advantages over commonly used DTW or LCSS TMMs. Its linear space and time complexity make it far less complex [19]. 2) Unlike some other TMMs, wLCSS eliminates the need for exclusive marking of gesture boundaries. This characteristic allows wLCSS to operate on a continuous IMU data stream that can be achieved by incrementing its value by reward *(R)* every time a match occurs between the incoming symbol stream (representing the pre-processed IMU data) and the template within the moving sliding window *(W)*. Conversely, we decrease it by a configurable penalty *(P)* for every mismatch encountered. Therefore, an AC gesture will be successfully identified when the match

score surpasses a specified threshold *(T)*. The purpose of the proposed preprocessing phase prior to applying wLCSS serves a bifold objective: firstly, to enhance robustness against noise, and secondly, to optimize template matching complexity, thereby reducing inference time.

1.3.6 Signal Energy Calculation

Following the implementation of the wLCSS technique, we added an additional phase aimed at computing the signal energy pertaining to each axis within the IMU data. This augmentation significantly enhances the robustness of the framework, as signal energy serves as a robust indicator of the signal's strength and intensity along each axis.

Further, for comparing and calculating each gesture template, we have implemented a majority voting mechanism. This involves combining wLCSS similarity results and signal energy calculations for each axis. The approach tallies the instances of a gesture being identified across axes and calculations, selecting the cough gesture with the highest votes as the final recognized one.

1.3.7 Coughalytics Smartwatch Application

We developed a custom real-time Android application our proprietary framework and conducted a comprehensive evaluation through a user study. Our application development employed Chaquopy, with the Android side operating in Java and the Python side utilizing Chaquopy in Android Studio. This integration allows for seamless interaction between the two programming languages, enhancing the overall functionality and performance of our smartwatch application.

Fig. 1.3 showcases the intuitive interface of our smartwatch application, featuring prominent start and stop buttons. Upon activation, the application guides users to capture two distinct gestures for each cough etiquette using these buttons. The initiation of the application automatically triggers the creation of a template library folder within the smartwatch's downloads directory. This designated

folder acts as a centralized repository for storing six gestures (two for each cough etiquette) as templates. These templates are later utilized by our application to calculate the mean values for each cough gesture, which are then compared with real-time datastreams. The application also not only detects coughs but also serves as a cough counter, tracking the occurrences of coughs and displaying the count on the screen.

Once all templates have been gathered, pressing the start button initiates the recording of the real-time datastream, enabling the detection of coughs matched against the templates. Concurrently, the application creates a microphone recordings folder on the smartwatch. When an AC gesture is detected, the microphone is activated for a duration of brief two seconds, capturing, and saving the recorded cough in this designated folder. Figure 3 illustrates an example of a recorded cough from one of the study participants. The two-second duration is designed to uphold privacy by only recording the cough event and promptly deactivating the microphone after recording.

Our smartwatch application is thoughtfully designed for compatibility with a wide range of wearOS smartwatches, supporting both older Android versions such as Android 6.0 (Marshmallow) and the latest Android 13.0 (Tiramisu). This broad compatibility ensures that our application can be utilized by a diverse user base, making it highly practical and accessible.

Overall, our Coughalytics framework achieves AC gesture detection on smartwatches, combining accelerometer and gyroscope data. It generates personalized templates for various cough etiquettes and employs a sliding window technique for real-time detection. The core utilizes wLCSS for efficient template matching, enhanced by signal energy calculation and majority voting. A custom Android application on Samsung Watch 4 facilitates user-friendly template collection, real-time detection, and recording of a cough.

Figure 1.3: Our customized smartwatch application interface illustrating the workflow for About-to-Cough (AC) detection, including few-shot template collection, template library organization, real-time data streaming, and on-device cough detection.

1.4 Experimenting Framework

To achieve this objective, we have undertaken an extensive study to evaluate the effectiveness of our framework. The comprehensive details of this investigation are thoroughly outlined in the following section. We conduct testing in-the-wild scenarios of the framework to assess its performance on a diverse set of individuals.

1.4.1 Gesture Analysis

To assess the performance of our framework, we conducted an in-depth analysis of cough gestures, encompassing all three etiquettes, alongside non-cough gestures referred to as null gestures. Within the null gesture category, we further classified it into two distinct groups: similar null gestures, such as itching face, eating, and drinking water; and dissimilar null gestures, which involve activities like using a laptop, opening doors, and operating a TV remote.

1.4.2 Participants

Four volunteers (2 self-identified females and 2 males, with an average age of 29.3±2.9) willingly took part in the testing of our framework. The Samsung Watch 4 was employed to collect data, capturing information from IMU sensors at a rate of 100 samples per second. Given that our framework was designed specifically for right-hand dominance, all participants were right-handed individuals. Each participant wore a smartwatch on their right hand for the duration of the study.

1.4.3 Procedure

All participants were equipped with fully charged smartwatches. They were exposed to diverse environments and instructed to execute cough gestures across all three etiquettes, as well as similar and dissimilar null gestures. Prior to testing, participants received training on using the application, including template collection and real-time gesture execution, enhancing user familiarity. We aimed for just two-shot learning for each cough etiquette (collectively six shots) during template generation to simplify the process for users. Additionally, a list of predefined gestures given in Table 1.1, such as eating, using a laptop, or drinking water, was provided to facilitate the testing process. The participants conducted their sessions in various real-world settings, allowing for a more natural evaluation. To maintain accuracy, we recorded videos of each volunteer performing various activities. After obtaining informed consent, videos of each participant performing various activities were recorded to establish accurate ground truth. The recordings were then reviewed to manually annotate the timing of each cough event for subsequent evaluation.

Table 1.1: Comprehensive list of activities recorded during the study, grouped into cough-similar hand gestures and non-similar daily activities for comparative analysis.

Similar hand gestures		**Non-similar hand gestures**	
1	Itching face	25	Sitting idle
2	Drinking water (glass/bottle)	26	Slow/Fast walking
3	Drinking coffee (mug)	27	Washing hands
4	Eating pasta (fork)	28	Wearing shoes
5	Eating spaghetti (fork)	29	Opening/closing door
6	Eating rice (spoon)	30	Opening/closing blinds
7	Eating banana	31	Toggle switch
8	Eating strawberry	32	Applying moisturizer (hands)
10	Eating soup (spoon)	33	Texting/scrolling on phone
11	Eating chips	34	Watering plants
12	Eating burger	35	Using TV remote
13	Eating sandwich	36	Cleaning table
14	Wearing/adjusting spectacles	37	Setting temperature (Google Nest)
15	Applying moisturizer (face)	38	Setting pillows
16	Applying lip balm	39	Lying on sofa
17	Brushing hair	40	Setting frames on wall
18	Tying hair	41	Opening/closing refrigerator door
19	Putting hair behind ear	42	Talking with hand gestures
20	Brushing teeth	43	Pulling tables
21	Wiping mouth (tissue)	44	Opening/closing curtains
22	Cleaning face (tissue)	45	Pouring water/coffee into a glass
23	Attending a call	46	Walking upstairs/downstairs
24	Yawning		

1.4.4 Ground Truth

To validate the accuracy of our Coughalytics application, we implemented a distinctive beep sound feature. This auditory cue was integrated to occur with the detection of coughs or null gestures. Whenever the application identified a cough or null gesture, a beep sound was emitted, indicating successful gesture recognition, and triggering a 2-second recording through the microphone.

For the purpose of establishing ground truth, we recorded videos of all participants. Subsequently, we analyzed these videos, pinpointing instances where the beep sounds occurred. This allowed us to differentiate between true positives (coughs) and false positives (null gestures). The count of identified coughs and null gestures provided a reliable foundation for assessing the application's accuracy.

Once the testing phase concluded, we made a user-centric decision to remove the beep sound from the application. This adjustment was implemented to enhance user experience by minimizing any potential annoyance associated with the auditory cue.

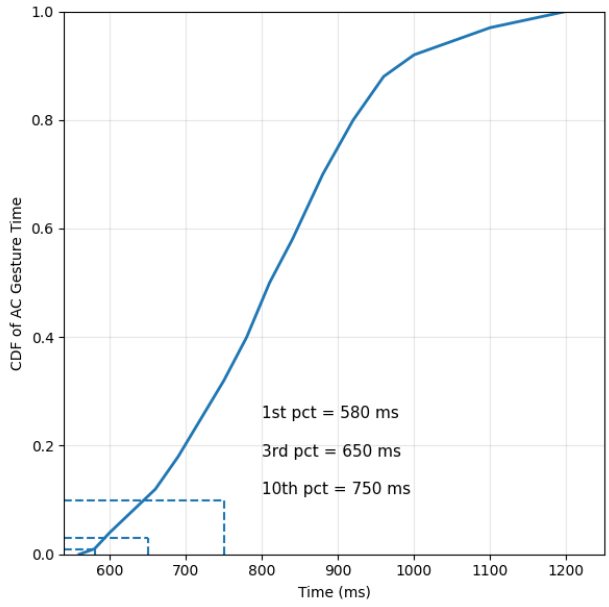

Figure 1.4: Cumulative distribution function (CDF) of About-to-Cough (AC) gesture duration, illustrating the time required to raise the hand or elbow to cover an incoming cough. Percentile thresholds (1st, 3rd, and 10th) highlight early response times across participants.

1.5 Results

1.5.1 Preliminary Feasibility of Meeting the AC Spotting Time Constraints

Based on the analysis of AC gesture IMU data shown in Fig. 1.4, the cumulative distribution function (CDF) of the time required to raise the hand or elbow to cover an oncoming cough indicates that early AC gestures occur within approximately 750 ms from the onset of the gesture. Therefore, the Coughalytics framework must complete AC gesture detection within this 750 ms timeframe to enable timely microphone activation before the cough event occurs.

The detection pipeline consists of four primary subtasks: 1) IMU data sampling, 2) pre-processing, 3) TMM inference, and 4) microphone activation. Empirical measurements summarized in Table 1.2 indicate that the AC gesture detection stage (sampling and inference) requires approximately 744 ms in total. Specifically, the smartwatch observes a 400 ms IMU sampling window, followed by approximately 344 ms required for pre-processing and TMM inference.

The smartwatch collects IMU data at a rate of 100 Hz, corresponding to 100 samples per second. Therefore, the number of samples collected within the sampling window is

$$N = f \times T = 100 \times 0.4 = 40$$

samples. This indicates that the framework obtains approximately 40 IMU samples before making a detection decision, which is sufficient to capture the temporal dynamics of the arm motion associated with the AC gesture. The 400 ms sampling window represents approximately

$$\frac{400}{750} \times 100 = 53.3\%$$

of the AC gesture duration, providing sufficient motion information for reliable detection.

Once the AC gesture is detected at approximately 744 ms after gesture onset, the smartwatch microphone is activated, requiring an additional 140.14 ms. Consequently, audio recording begins at approximately 884 ms after the start of the AC gesture, after which the microphone records audio for two seconds to capture the cough event.

Watch \ Time	Sampling (ms)	AC Gesture Spotting (ms)	Microphone Activation (ms)	Propagation Delay (ms)	Total Delay (ms)
Samsung Watch 4	400	344	140.14	0	884.14
Laptop	400	25	140.14	20	585.14

Table 1.2: Latency breakdown of the AC gesture detection pipeline. Detection occurs after sampling and inference, while microphone activation occurs afterward.

Our experimental results, shown in Fig. 1.5, indicate that using a short detection window of 300 ms leads to a substantially higher false positive rate (FPR). At this duration, the framework observes only a limited portion of the AC gesture, making it difficult to reliably distinguish cough-related arm movements from similar null gestures. As a result, many non-cough activities are incorrectly classified as AC gestures. Increasing the detection window provides more temporal information about the gesture dynamics, enabling more reliable discrimination between cough and non-cough movements. Based on this analysis, a 400 ms sampling window provides a practical balance between detection reliability and latency, supplying sufficient IMU samples for identifying salient AC gesture patterns while reducing the false positives observed with shorter detection windows.

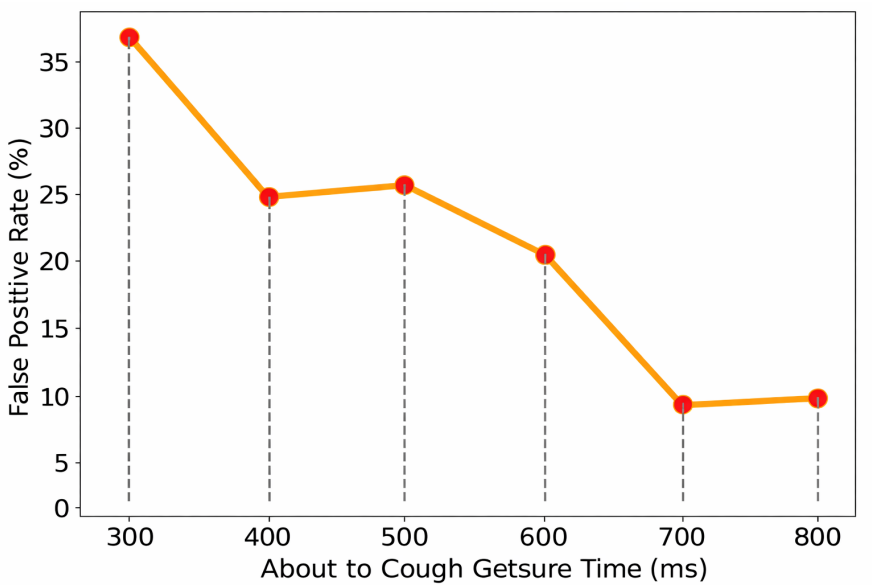

Figure 1.5: False positive rate (FPR) of About-to-Cough (AC) gesture detection as a function of the detection window size (ms). Increasing the AC window from 300 ms to 700 ms markedly reduces FPR.

1.5.2 All Participants Results

Our four participants wore the smartwatch continuously for a period of 5 hours in different independent environments, engaging with the application throughout. They executed both intentional and unintentional coughs, as well as null gestures, in a natural and unscripted manner. Following thorough testing on all participants, the application demonstrated an impressive overall average accuracy of 87% (±3.0) as shown in Table 1.3.

Furthermore, the application exhibited high sensitivity at 87.95% (±3.4) and specificity at 86.1% (±9.3). This indicates its robust ability to accurately identify true positives, showcasing its superior performance in detecting coughs over null gestures.

The exceptional level of accuracy was achieved by detecting coughs at an incredibly early stage, within a mere 744ms of their onset, when the cough gesture had only reached completion by 53%. The timeliness of our approach allows for the recording of crucial cough data via a smartwatch, affording ample time for subsequent processing. This early detection provides a valuable early

insight into cough occurrences, individuals and healthcare professionals are empowered to take proactive measures.

Table 1.3: Per-user detection performance for the About-to-Cough (AC) gesture. Confusion-matrix values (TP, FN, FP, TN) and derived metrics are reported in %.

Users	TP	FN	FP	TN	Accuracy	F1	Sensitivity	Specificity	Precision
User 1	90	10	13.5	86.5	88.2	88.4	90	86.4	86.9
User 2	83.3	16.7	16.4	83.5	83.4	83.4	83.3	83.5	83.5
User 3	87.5	12.5	15.7	84.2	85.8	86.1	87.5	84.2	84.7
User 4	91	9.0	9.6	90.3	90.6	90.7	91	90.3	90.3

TP = True Positive, FN = False Negative, FP = False Positive, TN = True Negative.

1.5.3 True Positives

In our study, the true positive rate pertained to accurately identifying genuine cough gestures as coughs. During our testing phase, we observed that the application consistently detected the majority of coughs with great precision. There were only a few instances where certain coughs were missed, typically due to variations in the cough gesture from the template or when coughs occurred simultaneously with other gestures, leading to potential confusion and categorization as a different gesture.

1.5.4 False Positives

In our scenario, false positives primarily occurred when the application incorrectly identified null gestures as cough gestures, despite no actual cough occurring. These false positives were most prevalent in situations where the gestures closely resembled coughing, such as when someone

was itching their face. However, it's worth noting that the results were notably impressive in distinguishing between coughs and various other similar actions, including eating, drinking water, cleaning the face with tissue paper, and brushing hair, among others.

1.5.5 Execution Time for Smartwatch

Developing an efficient real-time AC gesture detection framework necessitates a deep understanding and optimization of execution time $T_{\text{execution}}$. Execution time is the duration required for the system to process input data and generate an output, such as detecting a cough in this context. We have already highlighted the significance of early cough detection, which allows us to allocate sufficient time for processing and activating the microphone on the smartwatch. In our framework, the execution time for AC detection is currently measured at 744ms. This execution time can be further broken down into two components: inference time $T_{\text{inference}}$, which amounts to approximately 344ms, and AC gesture spotting time $T_{\text{AC Gesture}}$, which accounts for 400ms. In our framework, the execution time for AC detection is given by:

$$\begin{aligned} T_{\text{execution}} &= T_{\text{AC Gesture}} + T_{\text{inference}} \\ &= 400 \text{ ms} + 344 \text{ ms} \\ &= 744 \text{ ms} \end{aligned}$$

1.5.6 Execution Time for Laptop

Additionally, we also consider another scenario in which the smartwatch processor requires some time to process results, and similar processing occurs on a laptop. The laptop transfers 400ms of data to the smartwatch and receives the results via Bluetooth, allowing us to save even more time. When the inference step is offloaded to a laptop with greater computational capability, the AC gesture spotting time is estimated to decrease from approximately 344,ms on the smartwatch to

about 25,ms. Consequently, the AC execution time for the laptop configuration becomes:

$$\begin{aligned} T_{\text{execution}} &= T_{\text{AC Gesture}} + T_{\text{inference}} \\ &= 400 \text{ ms} + 25 \text{ ms} \\ &= 425 \text{ ms} \end{aligned}$$

1.5.7 Total Processing Time and Cough Event Coverage

After an AC gesture is detected, additional time is required to activate the smartwatch microphone before audio recording begins. For the Samsung Watch 4 used in our study, the microphone activation time is approximately 140.14 ms. Therefore, although the AC gesture is detected at approximately 744 ms after the onset of the gesture, the microphone becomes active at approximately 884.14 ms.

Previous studies analyzing cough acoustics indicate that the explosive phase of a cough sound typically lasts a few hundred milliseconds; however, the complete cough event, including airflow dynamics, acoustic decay, and possible sequential cough bursts within a cough bout, may extend to approximately 1–3 seconds depending on the individual and respiratory condition [20, 21]. Visual inspection of the recorded cough waveforms shown in Fig 1.6 also indicates that typical cough events observed in our dataset span roughly 1–2 seconds.

Assuming a conservative cough event duration of approximately 2 seconds from the onset of the AC gesture, the remaining portion of the cough event available for recording after microphone activation can be estimated as

$$2000 - 884.14 = 1115.86 \text{ ms}.$$

This indicates that the smartwatch captures approximately

$$\frac{1115.86}{2000} \times 100 \approx 55.8\%$$

of the cough event after recording begins. In addition, the smartwatch records audio for a fixed duration of 2 seconds after microphone activation, ensuring that the trailing portion of the cough waveform and surrounding respiratory acoustic context are fully captured.

In the laptop-assisted configuration, inference latency is reduced to approximately 25 ms. In this case, the AC gesture is detected at approximately

$$T_{\text{execution}} = 400\,\text{ms} + 25\,\text{ms} = 425\,\text{ms}.$$

Accounting for Bluetooth communication latency of approximately 20 ms and microphone activation time of 140.14 ms, audio recording begins at approximately 585.14 ms. Under the same 2 seconds cough event assumption, the remaining portion of the cough event available for recording becomes

$$2000 - 585.14 = 1414.86 \text{ ms},$$

corresponding to approximately

$$\frac{1414.86}{2000} \times 100 \approx 70.7\%$$

of the cough event being captured.

These results indicate that even with the computational constraints of an on-device smartwatch implementation, the system is capable of capturing a substantial portion of the cough event, while the laptop-assisted configuration further improves coverage by enabling earlier detection and recording.

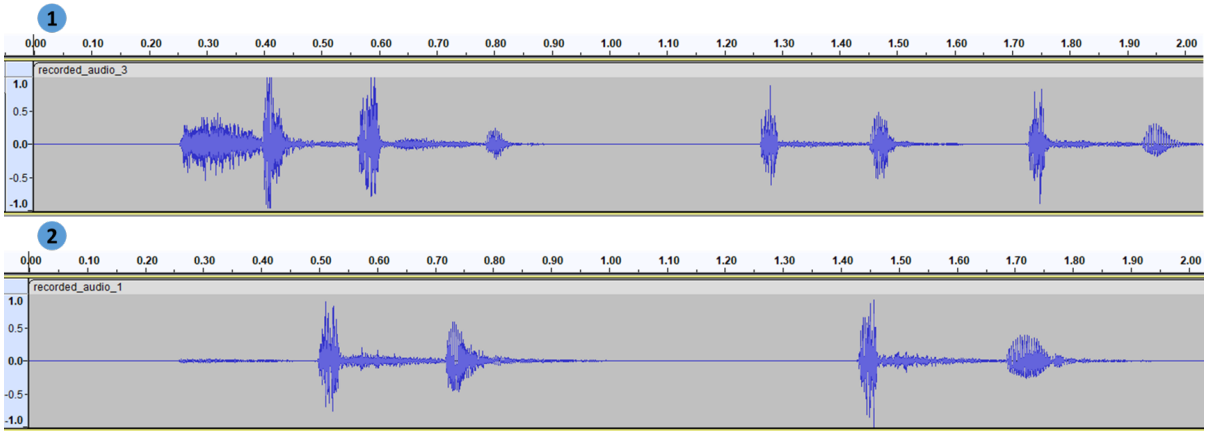

Figure 1.6: Representative cough recordings collected from study participants. The horizontal axis represents time (seconds), while the vertical axis denotes normalized amplitude on a linear scale from -1 to 1.

1.6 Applications

In this section, we have discussed various application examples to illustrate the versatility and utility of our application in diverse scenarios.

1.6.1 In home cough monitor

Coughalytics in an in-home environment can serve as a real-time respiratory health awareness tool, allowing individuals to monitor their cough within the comfort of their homes. This self-monitoring capability can facilitate early detection of respiratory issues and encourages a proactive approach to healthcare.

1.6.2 Healthcare physicians

Integrating Coughalytics into telemedicine platforms can enhance remote patient monitoring, enabling healthcare professionals to assess patient's respiratory health during virtual consultations.

This integration can provide a valuable tool for real-time health assessments and early detection of respiratory issues.

1.6.3 Clinical Research

Collaborating with researchers and pharmaceutical companies for clinical trials can allow the technology to play a crucial role in collecting data on participant's cough patterns and respiratory health. Coughalytics can contribute to advancing our understanding of respiratory conditions and refining treatment approaches.

1.6.4 Occupational Health and Safety

The implementation of Coughalytics application in occupational health and safety applications can provide continuous monitoring of workers in environments where respiratory health is a concern. This is particularly relevant in settings such as factories, construction sites, or areas with exposure to pollutants.

1.6.5 Public Health Monitoring

Collaboration with public health organizations can lead to the creation of a system for monitoring respiratory health trends in specific geographic areas. Coughalytics can assist in the early detection of potential outbreaks or environmental factors impacting respiratory health.

1.7 Conclusion and Discussion

In this chapter we present a groundbreaking development in the field of healthcare, our innovative smartwatch-based framework for early detection of coughs. We have successfully designed a

framework that not only detects coughs of three different etiquette (fist, elbow, and sleeve cough) but remarkably identifies them even before their completion, capturing coughs at a mere 53% of their progression. Our framework supports end-users to add their own customized gestures with very few samples, without impacting the recognition performance. We conducted our experiments on 4 different users in different environments with an exceptional performance of 87% (±3.0) accuracy rate in-the-wild scenarios. By harnessing the power of smartwatches, this technology can be seamlessly integrated into any Android device, empowering healthcare personnel and individuals alike to vigilantly monitor for signs of respiratory diseases. By facilitating the timely detection of coughs and other respiratory symptoms, it stands to revolutionize healthcare and disease management.

CHAPTER 2

InhaleSense: Monitoring Asthma to Detect Coughing and Inhaler Usage using Smartwatch Sensors

2.1 Introduction

2.1.1 Motivation

Asthma is a chronic respiratory condition characterized by recurrent episodes of wheezing sound in chest in addition to coughing, breathlessness, and chest tightness [22]. Some studies highlight the critical role of continuous cough frequency monitoring as a vital predictor of exacerbations asthma [2]. These exacerbations are preventable if predicted and treated timely. Recent studies show that among the many markers studied for COPD and asthma exacerbation prediction, cough frequency, if monitored continuously, is the most effective predictor [2, 23–26]. For example, clinical studies in [27, 28] show that even a simple always-on domiciliary audio recording-based cough counting tool alone can be used to predict (and thus prevent) the acute exacerbation of

COPD 3.4 days in advance and with a far lower false alarm rate than the widely used Leicester Cough Questionnaire (LCQ) or other daily questionnaires [4, 29–32]. The growing recognition of the clinical significance of monitoring and analyzing patient-specific cough has already led to regulatory approvals and commercialization of a few cough monitoring tools [33–44]. These tools essentially consist of an always-on microphone strapped to the chest or neck and a portable recording device [7, 8, 29, 35, 37, 38, 45–51]. Proprietary signal processing algorithms or trained humans then extract the cough count from all ambulatory sounds recorded during the monitoring period. However, none of the existing cough monitoring and analysis tools meets the privacy protection and form factor requirements for free-life wearability that is essential for unleashing this potential [52, 53].

Cough monitoring is particularly relevant for a specific subtype known as cough-variant asthma. Individuals with this type of asthma typically do not find relief from standard over-the-counter cough medications, instead, they require prescription asthma treatments, commonly administered through inhalers. These inhalers work by opening the airways in the lungs, providing rapid relief from coughing episodes, or by using a corticosteroid inhaler to reduce inflammation when used consistently as directed. Proper management of asthma symptoms is essential for improving patients' quality of life and preventing severe exacerbations. Researchers emphasize that it is crucial for asthma patients to use their inhalers exactly as prescribed by their doctors. Some studies showed that overuse of an inhaler may indicate poor asthma control and can lead to an increased risk of severe attacks and side effects such as increased heart rate, jitteriness, nervousness, and headaches. Conversely, underuse of the inhaler can reduce its effectiveness, leaving individuals more susceptible to asthma attacks and making it harder to manage symptoms during an episode, highlighting the importance of controlled and consistent inhaler use for asthma patients [54, 55].

As a practical approach to this pressing issue current state-of-the-art (SoTA) asthma detection systems employ a diverse range of machine learning (ML), deep learning (DL), and heuristic methods to classify respiratory features. Recent research has focused on asthma diagnosis, detection,

and management, leveraging multimodal sensors embedded in smartphones and smartwatches for respiratory condition monitoring [2]. Common approaches involve heuristic methods requiring users to position devices on specific body parts, such as the chest or abdomen, or utilize additional wearables like chest bands [56, 57]. Many existing methods that rely on ML models like random forests to classify respiratory behaviors using ultrasound data across different postures [56]. DL models, including 2D convolutional neural networks (CNNs), have been applied to cough-based respiratory disease diagnosis, analyzing features such as respiratory phase frequency, heart sounds, and breath sounds to differentiate asthma and COPD [57]. Sophisticated hybrid models that integrate U-NET architectures with LSTM networks have also been employed to differentiate between asthma and healthy groups by analyzing respiratory metrics obtained from finger-based pulse oximeters [58]. Additionally, inhaler usage classification has been explored using ML architectures like GMM, SVM, random forests, and AdaBoost, leveraging Bluetooth-enabled inhalers [59].

However, despite the high accuracy reported by these methods, critical challenges remain unresolved. Many studies do not sufficiently address the practicality and usability of these devices for everyday use; bulky devices or patches can be cumbersome, easily forgotten, or may not be readily accessible, particularly in the case of Bluetooth-based inhalers. Furthermore, the continuous monitoring of cough sounds raises significant privacy concerns, as audio recordings can also capture background noise along with cough sounds. On the other hand, despite the advent of smart inhalers [40–44, 60], poor compliance with the prescribed inhaler use remains a major factor in the exacerbation of asthma that leads to one death every four minutes in the USA alone [61]. These facts signify the dire need of a novel device that can: 1) monitor the state of a RD in asthma patients and predict its exacerbation well in advance with minimum false alarm rate through free-life cough monitoring and analysis; 2) automatically monitor the inhaler use and help increase inhaler use compliance through intelligent corrective feedback and only need-based reminders for missed inhaler use. 3) The device must preserve user privacy by not recording any speech, 4) and should be affordable to allow use by all.

Motivated by this unmet need and on our recent breakthrough in gesture recognition via smartwatches "Coughalytics", in this chapter, we propose an AI-enabled privacy-aware lightweight framework for screening and managing asthma. In summary, this chapter makes the following core contributions:

- We present a novel approach for Inhaler and cough detection using smartwatch accelerometer and gyroscope sensors data. Our model leverages hand gestures to predict cough and Inhaler gestures with high accuracy.
- By integrating cough detection with inhaler usage monitoring, our research addresses two critical aspects of asthma management, frequency of coughs and inhaler use. This dual-focus approach enhances symptom control and provides valuable insights into patient health.
- Our solution is designed to be lightweight and efficient, making it practical for implementation on smartwatch platforms and suitable for real-world use.
- In contrast to previous studies that rely on bulky or privacy-compromising methods, our study is designed to integrate smoothly into users' daily lives without the need for additional devices, ensuring ease of use and privacy.

This approach leverages smartwatches to seamlessly integrate cough detection and prediction into daily life, using hand gestures to monitor chronic conditions like asthma without compromising privacy. By enabling timely use of inhalers, this technology can reduce the severity and frequency of asthma attacks. In high-stakes scenarios like monitoring cough and inhaler usage, accuracy is critical, as false readings can lead to unnecessary or missed doses. To address this, we're developing lightweight, highly accurate gesture recognition models capable of real-time operation.

While existing gesture recognition research often lacks specificity and naturalness in real-world applications, our research focuses on accurately detecting hand gestures related to cough and inhaler use. By utilizing smartwatch sensors, we aim to create an intuitive, efficient solution that integrates

seamlessly into users' routines. This study introduces the first comprehensive framework for reliably detecting both inhaler use and coughs on smartwatches.

The chapter is organized as follows: Section 2.2 outlines our proposed methodology, including data acquisition, detailed model architecture, and the initial setup for model training and testing. Section 2.3 delves into model refinement and the data augmentation techniques employed in our work. Finally, Section 2.4 provides a detailed evaluation and performance analysis of our model.

2.2 Proposed Methodology

We developed a system composed of two model components trained offline using the collected dataset. In this section, Section 2.2.1 describes the data collection process, Section 2.2.2 presents the model architecture and its components, and Section 2.2.3 details the model training procedure and performance evaluation.

2.2.1 Data Collection for Model Training

In this study, IMU data were collected from smartwatches to capture cough and inhaler usage gestures using accelerometer and gyroscope sensors. Data were recorded using a Samsung Galaxy Watch 4 at a sampling rate of 100 Hz. The dataset was collected from five participants (3 self-identified females and 2 self-identified males) with a mean age of 28.4 ± 3.0 years to evaluate the feasibility of a smartwatch-based gesture recognition framework for asthma management.

A custom smartwatch application was developed to facilitate data collection using linear acceleration and gyroscope sensors. The application provides simple start and stop buttons to initiate and terminate data recording, and all recordings are automatically saved to the smartwatch downloads folder. Data collection was performed in a routine setting, where cough and inhaler usage gestures were recorded separately for training purposes.

To establish a controlled baseline for gesture recognition, the training data were collected from non-asthmatic individuals who mimicked inhaler usage gestures. This approach provides a clear reference for ideal inhaler usage patterns while avoiding the need for early-stage data collection from asthmatic patients.

To evaluate real-time performance, testing was conducted with the same participants performing coughing and inhaler gestures alongside random daily activities representing non-cough and non-inhaler actions. With participants' consent, videos were recorded using their mobile phones to capture the exact timing of coughing and inhaler gestures. These recordings were used to generate ground-truth annotations for accurate evaluation. The primary objective of this work is to develop a robust machine learning model capable of accurately detecting gestures associated with asthma management.

Gesture Data

The proposed framework focuses on two gesture categories relevant to asthma management: coughing behaviors and inhaler usage gestures. Coughing is a common symptom associated with asthma, and cough-related gestures often involve distinct hand movements that can be captured through wrist-worn IMU sensors. In this study, we considered four coughing behaviors [17, 18]: fist cough, elbow cough, sleeve cough, and mouth-cover cough.

In addition to cough monitoring, inhaler usage gestures were recorded to capture medication-related behavior. The inhaler gesture sequence includes movements associated with shaking the inhaler, positioning it, and performing the inhalation step. Monitoring both coughing and inhaler gestures enables the system to identify behaviors associated with asthma symptom management.

To illustrate these gestures, Fig. 2.1 presents representative IMU signal segments alongside gesture illustrations. In total, more than 400 cough and inhaler gesture instances were collected for model training and evaluation.

Null Data

To ensure robust gesture recognition, additional data were collected for non-target activities representing everyday movements that may resemble cough or inhaler gestures. These activities include drinking water, yawning, running, answering phone calls, and other routine hand movements. A total of more than 500 null gesture instances were collected, allowing the model to distinguish asthma-related gestures from common daily activities.

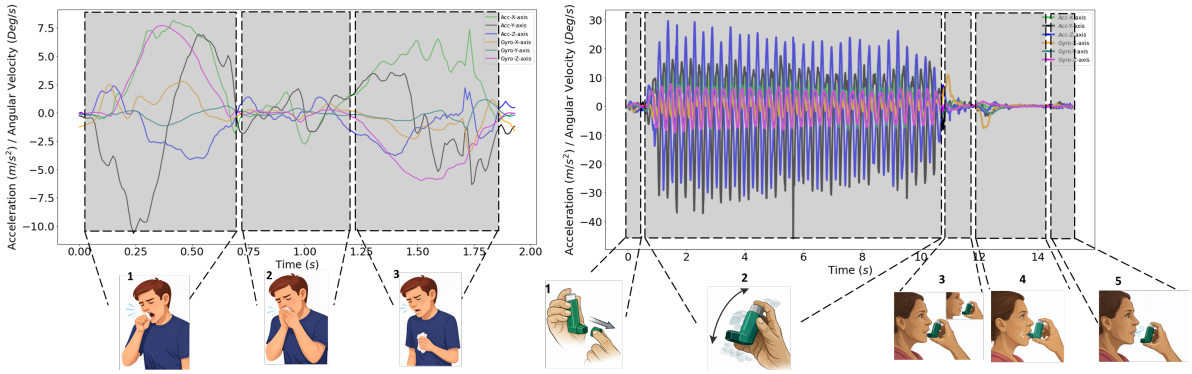

Figure 2.1: Segmented three-axis smartwatch IMU time-series signals for (left) cough and (right) inhaler-use gestures. The dashed boxes indicate temporally partitioned phases of each activity, with corresponding gesture illustrations shown below to highlight motion transitions across segments.

2.2.2 Model Architecture Overview

In this section, we present an overview of the proposed InhaleSense architecture for inhaler management classification, as illustrated in Fig. 2.2. The framework employs EfficientNet as the backbone network, incorporating Mobile Inverted Residual (MBConv) blocks for efficient feature extraction. EfficientNet's lightweight design enables high performance while maintaining a low number of parameters and reduced computational cost compared to traditional architectures, making it well suited for on-device deployment.

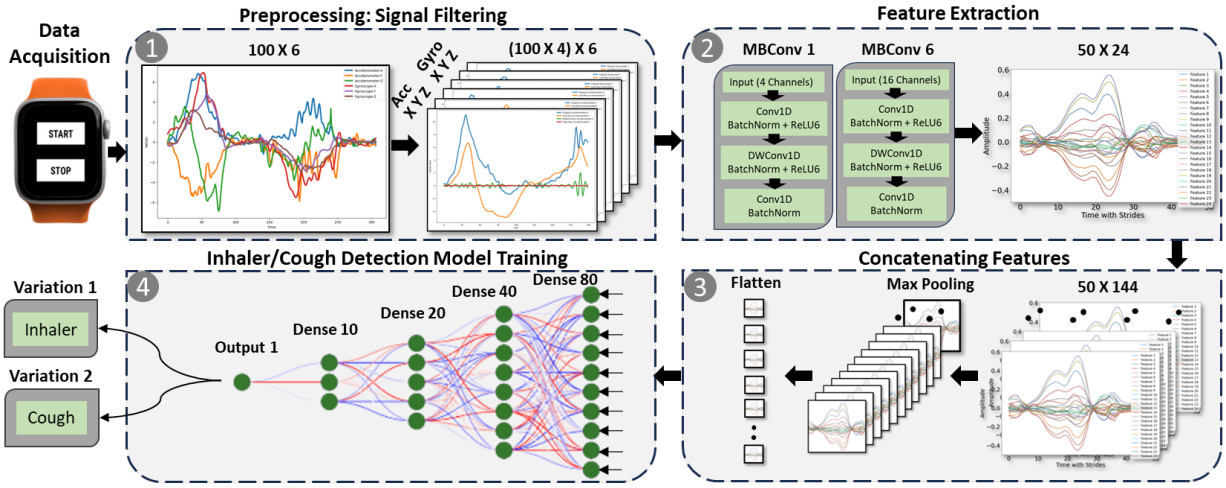

Figure 2.2: Following data collection, the framework is organized into four main stages. Part I consists of two subcomponents: (i) signal filtering and (ii) feature extraction. Part II further comprises (iii) feature concatenation and (iv) gesture training and detection.

Preprocessing (Signal Filtering)

After data acquisition, preprocessing is performed to clean the signals and handle missing values. To ensure accurate training on cough and inhaler gestures, silent periods at the beginning and end of each sequence are removed.

Further preprocessing involves applying three Butterworth band-pass filters to capture different frequency components of the motion signals. The filters target three frequency ranges: low (0.22–8 Hz), medium (8–32 Hz), and high ($>$ 32 Hz), implemented using cascaded second-order sections. This filtering process extracts motion characteristics across multiple frequency bands, improving the representation of gesture dynamics.

The IMU data consists of measurements from three accelerometer axes and three gyroscope axes. For each axis, the filtered signals generate additional frequency-band features. Consequently, each channel is represented by four features (one raw signal and three filtered components) over a temporal window of 100 samples, resulting in a feature dimension of 100×4 per channel.

Model Part 1 (Feature Extraction)

To balance model complexity and performance, the framework utilizes an EfficientNet-inspired architecture based on Mobile Inverted Residual (MBConv) blocks for feature extraction. Specifically, MBConv1 and MBConv6 blocks are employed to efficiently capture motion patterns from the processed IMU signals.

The MBConv1 block uses depthwise separable convolutions to reduce computational cost while maintaining representational capacity. Two MBConv1 layers with stride 1 are applied to preserve the spatial dimensions of the feature maps while extracting initial motion features.

To capture more complex patterns, the MBConv6 block is introduced. This block expands the feature maps using an expansion factor of six and applies a stride of 2 to reduce spatial dimensions while increasing representational depth. For each channel, the four preprocessing features are expanded into 24 feature maps (4×6) through the MBConv6 operation.

ReLU6 activation is applied within both MBConv1 and MBConv6 blocks to introduce non-linearity and improve the model's ability to learn complex gesture patterns. After processing individual channels, the outputs from all six IMU channels are concatenated to form a unified feature representation. A separable convolution layer is then applied to integrate cross-channel information efficiently, followed by a max-pooling layer to downsample the feature maps. Finally, the resulting features are flattened to prepare them for the fully connected classification layers.

Model Part 2 (Inference)

The flattened feature representation is passed through a sequence of five fully connected layers with 80, 40, 20, 10, and 2 neurons, respectively. Batch normalization and dropout (rate = 0.5) are applied between consecutive layers to improve generalization and stabilize the training process.

The final output layer produces confidence scores for two classes: either *cough* vs. *null gesture*

or *inhaler* vs. *null gesture*. Model training uses the cross-entropy loss function with the Adam optimizer for efficient parameter updates.

Despite its expressive capability, the model remains compact, containing only 33,128 parameters (approximately 129.41 KB). This lightweight architecture enables efficient on-device inference, allowing real-time gesture recognition without requiring significant computational resources.

2.2.3 Model Training and Performance

To demonstrate the effectiveness of cough and inhaler gesture recognition and evaluate the robustness of the InhaleSense framework, we created separate training datasets for cough and inhaler gestures. In each dataset, the number of cough or inhaler gestures was balanced with null class gestures to address the class imbalance problem. After training the models using these balanced datasets, separate sets of unseen testing data were generated for evaluation.

For the pretrained model, we collected over 400 cough samples, 400 inhaler usage gesture samples, and more than 500 null samples consisting of gestures dissimilar to coughing and inhaler use. To address variability in null data, the training process was designed to create two distinct model variations: a cough-based model and an inhaler-based model. The cough-based model is trained to classify coughing gestures while identifying any other gesture as non-cough. Conversely, the inhaler-based model focuses on recognizing inhaler gestures and classifies all other gestures as non-inhaler. This dual-model approach is necessary because the duration of each gesture differs significantly: the average cough gesture lasts approximately 2 seconds, whereas the average inhaler gesture lasts around 12 seconds. The proposed model is explained in detail in the next subsection.

Cough Training Data

For training the InhaleSense model, two-second sequences of six-degree-of-freedom accelerometer and gyroscope data are processed for cough detection. This duration is based on the typical length

of a cough gesture, which lasts approximately 2 seconds according to the collected participant data. The cough gesture sequence includes the initial hand movement preceding the cough and the subsequent return motion. The cough training dataset contains 400 cough gesture instances and 500 null gesture instances to maintain approximate class balance.

Inhaler Training Data

For the inhaler model, twelve-second sequences are processed to capture inhaler usage gestures. According to established inhaler usage procedures, the gesture typically includes shaking the inhaler for approximately 10 seconds followed by positioning and inhalation, resulting in an average total duration of about 12 seconds, as illustrated in Fig. 2.1. The inhaler usage sequence therefore captures the complete motion pattern from shaking the inhaler to the final inhalation, distinguishing it from other routine hand gestures.

Model Training

We trained the InhaleSense model on the data distributions outlined earlier in this section to evaluate its performance. The models were implemented using Python 3.6 and TensorFlow 1.21, with the training process accelerated using a high-performance computing cluster.

To achieve optimal accuracy, we carefully selected the hyperparameters for both model variations. A grid search strategy was used to determine the optimal values for the learning rate, number of epochs, batch size, optimizer, dropout rate, and the number of layers and nodes. This process involved grid search with cross-validation, where the dataset was randomly divided into training, validation, and testing sets. Specifically, the data was first split into a training set (90%) and a test set (10%). The training set was then further divided into a final training set (81%) and a validation set (9%). Each model was trained on the training set using multiple hyperparameter configurations, and performance on the validation set was used to determine the best configuration.

Metrics such as sensitivity and specificity from the validation set over 100 epochs were used to identify the optimal hyperparameters. After selecting the best configuration through grid search, the final training phase was performed for both models. Table 2.1 lists the optimal hyperparameter settings for the inhaler and cough models. The cough and inhaler models were then trained using these settings on their respective training sets and evaluated on unseen test sets to ensure an unbiased assessment of performance.

Table 2.1: Details of hyperparameters used for the InhaleSense Framework after optimization.

Learning rate	Number of epochs	Batch size	Optimizer	Dropout rate	Activation function for feature extraction in MBConv	Activation function for training in FC layers	Sliding window for cough (Seconds)	Overlap in sliding window for cough (Seconds)	Sliding window for inhaler (Seconds)	Overlap in sliding window for inhaler (Seconds)	Random seed	Max pooling pool size	Number of FC layers	Total number of parameters
0.001	100	32	Adam	0.5	ReLU6	ReLU	2	0.5	1200	300	42	8	5	33,128

MBConv = feature extraction in MobileNet-style blocks; FC = fully connected. Sliding window and overlap values in seconds unless noted.

Model Performance on Initial Setup

After data collection and hyperparameter selection, each data sequence was processed using a sliding window mechanism. The window size was set to 12 seconds for inhaler gestures and 2 seconds for cough gestures, corresponding to the input size of the model.

We first examined the predicted outcomes at the window level. The model performed exceptionally well for inhaler gestures, almosty accurately distinguishing them from similar null gestures due to their distinct motion patterns. However, the performance for cough gesture detection showed an average accuracy of 74.8%.

Further analysis revealed that the model frequently confused cough gestures with null gestures. Most misclassifications occurred when the model failed to detect cough windows where a gesture was actually present, resulting in false negatives. This issue arose because the 2-second window captured only a narrow segment of the gesture, often including portions of null activity at the beginning or end. Consequently, correct predictions were achieved only when the window precisely aligned with the trained cough gesture segment.

In real-time scenarios, gestures occur in a continuous stream and may span multiple windows. To address this issue, a sliding window technique was applied, allowing partial overlap between consecutive windows and incorporating information from preceding segments. Fig. 2.3 illustrates that cropping the cough gesture data to 0.3 seconds significantly degraded the performance to 30%, and further cropping to 0.4 seconds resulted in additional deterioration in window-level predictions on the testing set.

To further improve robustness, the dataset was augmented by increasing both null and cough gesture samples. Augmentation techniques included cropping gesture segments and inserting null data at the beginning or end of gesture samples. This strategy was intended to improve the model's ability to recognize partial gestures and better distinguish between cough and non-cough activities.

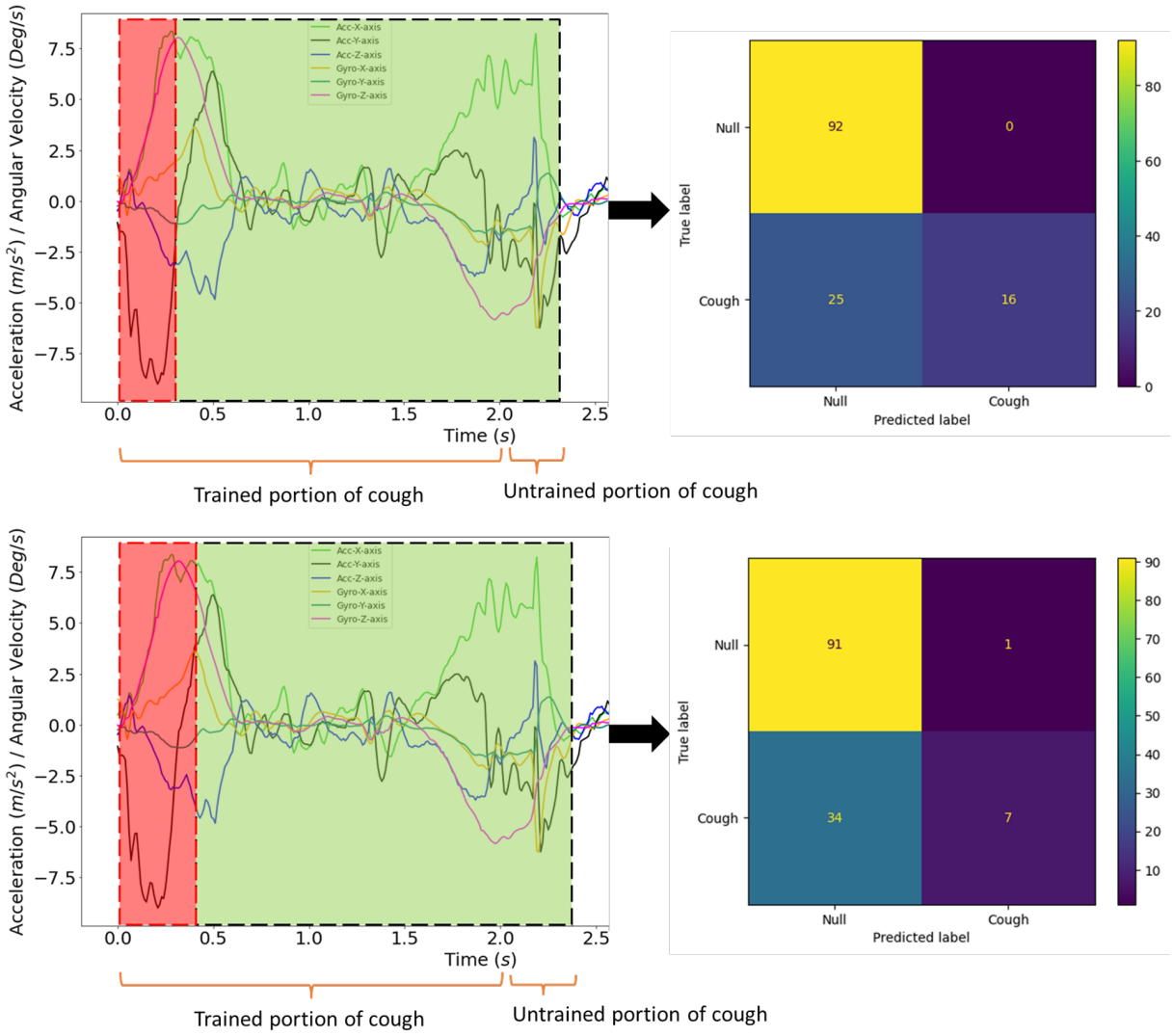

Figure 2.3: Impact of early temporal cropping on cough detection performance. Cropping the training data to the first 0.3 s (top) and 0.4 s (bottom) of the cough signal results in noticeable performance deterioration, highlighting the importance of sufficient temporal context for robust classification.

2.3 Model Refinement and Data Augmentation

Training a model with a limited number of samples presents significant challenges, particularly in preventing overfitting and reducing false positives. With a small dataset, the model may struggle to generalize effectively because it lacks sufficient positive gesture samples to learn representative

patterns. To address this limitation, we implemented several data augmentation strategies to maximize the utility of the available user data. One key technique was a sliding window approach that segmented gesture sequences into overlapping windows. This method expanded the effective dataset size and enabled the model to learn gesture patterns even when only partial motion information was present. These strategies, together with the model architecture, form the complete InhaleSense framework illustrated in Fig. 2.2.

2.3.1 Overlapping Sliding Window Technique

For the time-series IMU data, gesture sequences were initially divided into fixed windows of 2 seconds for cough gestures and 12 seconds for inhaler gestures. However, a simple non-overlapping windowing strategy proved insufficient for accurately capturing cough gestures. Due to the short duration of cough events, the gesture could easily be missed if the window boundaries did not align with the motion pattern.

To address this limitation, an overlapping sliding window technique was employed, where a portion of the previous window is retained in the subsequent window. The overlap ratio was treated as a hyperparameter and tuned to optimize detection performance. If the overlap is too large, the model may detect the same gesture multiple times, leading to false positives. Conversely, if the overlap is too small, the model may fail to capture sufficient contextual motion information, increasing the likelihood of missed detections.

Through extensive experimentation, an overlap of 25% was found to provide the most reliable results for both cough and inhaler gestures. This overlap improved the model's ability to capture gesture transitions and detect cough events more consistently. However, overlapping windows introduce the possibility of detecting the same gesture multiple times. To mitigate this issue, we introduced a mechanism termed the *Guard Interval*. The Guard Interval acts as a temporal buffer that temporarily suppresses new detections after a gesture has been identified.

For cough gestures, the guard interval was set to 2 seconds, while for inhaler gestures it was set to 12 seconds, matching the duration of their respective detection windows. Within this interval, it is unlikely that the same gesture will occur again, so the model ignores similar signals during this period. This mechanism prevents repeated detections caused by overlapping windows and ensures that each gesture instance is counted only once in a continuous streaming environment. As a result, the proposed system becomes more robust, accurately detecting distinct gesture events while minimizing false positives.

2.3.2 Data Augmentation Techniques

To improve the model's performance, particularly for cough gesture detection, several data augmentation techniques were applied to generate a larger and more diverse set of samples. The primary challenge was the accurate detection of cough gestures, which required increasing both the quantity and variability of cough data to prevent the model from overfitting to specific gesture boundaries. This overfitting was observed when the model performed well only when real-time cough gestures closely matched the training sequences.

Sliding Window Augmentation

We implemented a sliding window augmentation strategy that randomly slides over half of the cough gesture sequences at different positions. This method generates additional training samples by capturing different segments of the cough gesture. In real-time scenarios, the model may encounter only partial segments of a gesture rather than the full sequence. By training the model on these partial segments, the system becomes more capable of recognizing cough gestures even when only a fragment of the motion pattern appears in the input. This approach reduces the model's reliance on exact start and end points, thereby mitigating overfitting.

Pre-pending Null Segments

In addition to sliding window augmentation, we introduced variability by prepending null segments to cough gestures. This technique involves adding portions of null gesture data to the beginning of cough sequences. The goal was to vary the onset of cough gestures so the model learns to distinguish genuine coughs from similar hand movements that may occur in real-time scenarios. By altering the starting point of the gesture, the model becomes less dependent on fixed temporal boundaries, improving its generalization capability.

While augmenting cough data was essential, increasing the diversity of null gesture data was equally important. The model must accurately differentiate between cough and null gestures, since confusion between them could result in false positives or missed detections. To address this, several augmentation techniques were applied to null gesture samples.

Masking

Masking was applied by randomly setting portions of the null gesture signal to zero at three different positions. This process exposes the model to varied null gesture patterns and improves its ability to distinguish between actual gestures and background motion.

Jittering

Jittering was performed by adding small random noise with a standard deviation of 0.01 to the null gesture signals. This introduces slight perturbations that simulate variations present in real-world motion data. As a result, the null gesture dataset becomes more diverse, reducing the likelihood of overfitting and improving the model's generalization to unseen signals.

It is important to note that these augmentation techniques were applied only to cough and null gesture data. The inhaler gesture already exhibited highly distinctive motion patterns and achieved

strong detection performance. Applying additional augmentation to this class could introduce unnecessary variability and potentially degrade the model's ability to recognize inhaler gestures accurately.

2.4 Model Evaluation and Performance

To evaluate the performance of the inhaler and cough gesture models, we conducted a user study designed to assess their ability to recognize both target gestures and other similar hand movements. The study evaluates the robustness of the models in distinguishing cough and inhaler gestures from a variety of real-world activities.

2.4.1 Data Collection for Testing

Real-time data were collected from the same participants in a natural and unconstrained environment. The evaluation focused on four cough etiquette gestures for the cough model and inhaler usage gestures for the inhaler model. To further test the model's robustness, additional gestures were included from a taxonomy of dynamic hand gestures to represent a diverse range of hand movements, including walking, cleaning, making coffee, playing, and other routine activities.

The objective of this gesture set was to evaluate the framework's ability to accurately recognize target gestures while distinguishing them from other everyday hand movements. Although the gesture set used in this study is not exhaustive, the framework is designed to support a broader range of gestures beyond those tested. All participants volunteered for the study and were right-handed. Each participant wore the Samsung Galaxy Watch 4 on their non-dominant hand. The smartwatch, equipped with IMU sensors sampled at 100 Hz, was used for real-time data collection.

To obtain reliable ground-truth annotations, participants recorded videos of their gesture activities using their own mobile phones after providing consent. Participants also provided timestamps

indicating when cough or inhaler gestures occurred. These timestamps were cross-referenced with the corresponding IMU data recorded by the smartwatch to accurately label gesture events. The data collection process was designed to remain unobtrusive, allowing participants to wear the smartwatch during their normal daily activities without disruption. Table 2.2 summarizes participant information, including age, gender, duration of smartwatch usage, and the number of cough and inhaler gestures performed.

Table 2.2: Demographic information and data collection statistics of participants used for model evaluation.

Participant	Age	Gender	Duration	Cough	Inhaler
1	29	Female	4h 30m	42	7
2	28	Female	1h	7	2
3	75+	Female	1h 10m	4	0
4	31	Male	1h 15m	10	0
5	26	Male	1h	5	1

2.5 Results

After collecting and labeling the data, we trained the models using the prepared dataset. For the cough model, each instance of a cough gesture was labeled accordingly, while all other gestures, including inhaler gestures, were classified as null. Similarly, for the inhaler model, inhaler gestures were labeled as the target class, while cough and other gestures were treated as null gestures.

To enhance model performance, both collected and augmented datasets were used during training. The training process incorporated both simple sliding window and overlapping sliding window techniques. The simple sliding window provided basic segmentation of the data, while the overlapping sliding window allowed the model to learn from adjacent temporal segments. This combination

improved gesture recognition accuracy while reducing the likelihood of false positives.

Model Performance for Inhaler Gesture Detection (Variation 1)

The inhaler gesture detection model demonstrated excellent performance, achieving approximately 99% accuracy across all participants. The distinct motion pattern associated with inhaler usage enabled the model to reliably differentiate inhaler gestures from other daily hand movements.

Model Performance for Cough Gesture Detection (Variation 2)

The cough gesture detection model also achieved strong performance despite the challenges posed by the wide range of gestures that may resemble coughing motions. The model successfully detected most cough events across participants, achieving an average accuracy of approximately 98%. This result highlights the model's ability to distinguish cough gestures from other common hand movements in real-world scenarios.

Detailed participant-level performance metrics are presented in Table 2.3 and Table 2.4, demonstrating the robustness of the proposed system for both inhaler and cough gesture detection.

Table 2.3: Performance evaluation of the proposed model for real-time cough detection. Per-participant detection metrics: sensitivity, false negative rate, specificity, false positive rate, and accuracy (%).

Participants	Sensitivity	False negative	Specificity	False positive	Accuracy
1	90.48	9.52	99.17	0.83	98.85
2	85.71	14.29	98.60	1.40	98.42
3	75.00	25.00	99.30	0.70	99.08
4	86.67	13.33	99.07	0.93	98.82
5	100.00	0.00	97.96	2.04	98.00

Table 2.4: Performance evaluation of the proposed model for real-time inhaler-use detection. Per-participant detection metrics: sensitivity, false negative rate, specificity, false positive rate, and accuracy (%).

Participants	Sensitivity	False negative	Specificity	False positive	Accuracy
1	100.00	0.00	100.00	0.00	100.00
2	100.00	0.00	88.89	11.11	90.91
3	N/A	N/A	N/A	N/A	N/A
4	N/A	N/A	N/A	N/A	N/A
5	100.00	0.00	100.00	0.00	100.00

2.5.1 Participant-Level Performance

To further analyze the effectiveness of the proposed system, we evaluated the model performance for each participant individually. The evaluation considered cough gesture detection and inhaler gesture detection using real-time data collected in natural environments. The metrics reported include sensitivity, false negative rate, specificity, false positive rate, and overall accuracy.

Participant 1

Participant 1 (female, age = 29) wore the smartwatch continuously for approximately 4 hours while performing daily activities such as working, cooking, and cleaning. During this session, the participant coughed 42 times and used the inhaler 7 times at random intervals. The cough gestures were evenly distributed across the four etiquette styles: 11 fist coughs, 11 elbow coughs, 10 sleeve coughs, and 10 mouth-cover coughs.

The cough detection model correctly identified 38 of the 42 cough events, resulting in a sensitivity of 90.48% with a false negative rate of 9.52%. The specificity achieved was 99.17% with a false positive rate of 0.83%, leading to an overall detection accuracy of 98.85%.

For inhaler detection, all inhaler gestures were correctly identified, resulting in 100% sensitivity, 100% specificity, and an overall accuracy of 100%.

Participant 2

Participant 2 (female, age = 28) wore the smartwatch for approximately 1 hour while performing daily activities such as making coffee, cleaning utensils, wearing a mask, and preparing vegetables. During this session, the participant coughed 7 times and used the inhaler twice. The cough gestures included 2 fist coughs, 2 elbow coughs, 2 sleeve coughs, and 1 mouth-cover cough.

The model detected 6 of the 7 cough gestures, resulting in a sensitivity of 85.71% and a false negative rate of 14.29%. The specificity achieved was 98.60%, with a false positive rate of 1.40%, producing an overall accuracy of 98.42%.

For inhaler gesture detection, both inhaler events were correctly detected. However, one false positive occurred during a similar hand movement while cleaning a coffee mug, resulting in a specificity of 88.89% and an overall accuracy of 90.91%.

Participant 3

Participant 3 (female, age = 75+) wore the smartwatch for approximately 1 hour and 10 minutes while performing activities such as drinking juice, playing board games, writing on a whiteboard, and conversational hand gestures. During this session, the participant coughed 4 times and did not perform any inhaler gestures.

The cough detection model correctly identified 3 of the 4 cough events, yielding a sensitivity of 75.00% and a false negative rate of 25.00%. The model achieved a specificity of 99.30% with a false positive rate of 0.70%, resulting in an overall accuracy of 99.08%.

Since no inhaler gestures were performed during this session, inhaler detection metrics were not

computed for this participant.

Participant 4

Participant 4 (male, age = 31) wore the smartwatch continuously for approximately 1 hour and 15 minutes while performing daily activities including typing, answering phone calls, walking quickly, and climbing stairs. During this session, the participant coughed 15 times at random intervals.

The cough detection model correctly identified 13 of the 15 cough events, achieving a sensitivity of 86.67% and a false negative rate of 13.33%. The model achieved a specificity of 99.07% and a false positive rate of 0.93%, resulting in an overall accuracy of 98.82%. No inhaler gestures were performed by this participant during the testing session.

Participant 5

Participant 5 (male, age = 26) wore the smartwatch for approximately 1 hour while performing daily activities such as typing, answering phone calls, walking quickly, and climbing stairs. During this session, the participant coughed 5 times and used the inhaler once.

The cough detection model successfully detected all cough events, achieving a sensitivity of 100.00% and a false negative rate of 0.00%. The specificity was 97.96% with a false positive rate of 2.04%, resulting in an overall accuracy of 98.00%.

The inhaler gesture was also correctly detected, yielding 100% sensitivity, 100% specificity, and an overall accuracy of 100%.

2.6 Discussion

In this section, we discuss the broader implications of our framework, including potential application scenarios, extensibility to additional gesture sets, and the overall generalizability of the proposed system. We also outline current limitations and directions for future research.

Applications

The proposed smartwatch-based framework has significant potential for improving asthma management and advancing wearable healthcare technology. By accurately detecting both cough and inhaler usage gestures through wrist-worn IMU signals, the system provides a dual-function solution for monitoring symptoms and medication adherence. This capability can assist patients in better understanding their condition and maintaining consistent inhaler usage, thereby supporting improved self-management of asthma.

The lightweight and privacy-aware design of the system enables seamless integration into everyday life. Unlike intrusive monitoring approaches that require additional sensors or bulky equipment, the proposed method operates using a standard smartwatch, preserving user comfort and privacy. This design makes the framework well suited for long-term continuous monitoring and encourages sustained use in real-world environments.

The framework is also well positioned to support remote healthcare and telemedicine applications. Clinicians can access objective data on cough frequency and inhaler usage without requiring frequent in-person visits. Such remote monitoring capabilities can improve clinical decision-making, enable earlier interventions, and reduce healthcare system burden.

Beyond Gestures

Beyond asthma management, the proposed gesture-recognition framework can potentially be applied to other healthcare and wearable sensing applications. Because the underlying architecture and signal-processing pipeline are largely independent of a specific gesture type, the system can be extended to other time-series recognition tasks involving wrist motion.

For example, the framework could be adapted to monitor medication adherence behaviors for other chronic diseases or to track rehabilitation exercises in physical therapy programs. Similarly, the system could support wellness monitoring tasks that involve repeated hand or arm movements. This flexibility highlights the broader applicability of the proposed approach for remote healthcare, assistive technologies, and personalized health monitoring systems.

Limitations

Despite the promising results, several limitations remain. First, the current gesture set is relatively limited. Although the evaluation includes multiple cough styles and various null gestures, the range of tested gestures is not exhaustive. Future work will expand the dataset by incorporating additional null gestures and conducting larger-scale user studies.

Second, the cough gesture model was trained using a relatively small and limited dataset. Increasing the diversity of cough and inhaler gesture samples would improve the model's ability to generalize to unseen users and reduce potential misclassifications with visually similar hand movements.

Third, the current system performs model training off-device using an external computer. While the model is lightweight, fully integrated on-device training and inference on smartwatch hardware remains outside the scope of this chapter. Future work will explore deploying the system directly on wearable devices or leveraging cloud-assisted processing for devices with limited computational resources.

Finally, the current framework detects inhaler usage events but does not evaluate whether the inhaler was used correctly. Future extensions could analyze additional motion patterns to verify correct inhaler handling, such as shaking duration or inhalation timing, enabling more comprehensive feedback for asthma management.

2.7 Conclusion

This chapter introduced a gesture-based customization framework for monitoring cough and inhaler usage using smartwatch-based IMU time-series data. To develop the system, we conducted an initial user study ($N = 5$) to train an IMU-based deep learning gesture recognition model capable of identifying asthma-related gestures from wrist motion.

Initial experiments showed strong performance for inhaler gesture recognition, achieving near-perfect detection accuracy. However, cough gesture detection proved more challenging due to the variability and similarity of cough movements with other daily activities. To improve robustness, several data augmentation strategies and overlapping sliding window techniques were applied to expand and diversify the training dataset.

In a subsequent real-time evaluation study ($N = 5$), the proposed framework demonstrated strong performance across participants. The cough detection model achieved an average sensitivity of 87.6% with an overall detection accuracy of 98.6%. For inhaler gesture detection, the system achieved an average accuracy of 96.97% across participants who performed inhaler actions.

These results demonstrate the feasibility of using smartwatch-based IMU sensing for real-time monitoring of asthma-related gestures in natural environments. More broadly, the proposed framework enables users to extend beyond predefined gesture sets and potentially train personalized gesture models for health monitoring and assistive wearable applications.

CHAPTER 3

Temporal Convolutional Networks with RoPE-Enhanced BERT: A Unified Approach to Multi-Modal Gesture Recognition

3.1 Introduction

3.1.1 Motivation

Body-focused repetitive behaviors (BFRBs) are among the most common yet under-recognized mental health conditions in both clinical and community settings [62, 63]. These behaviors are often driven by stress relief, boredom, or gratification, although a substantial proportion of individuals report no clear motivating factor and express ambivalence toward their actions [64]. These behaviors include actions such as skin picking, hair pulling, and nail biting [65]. While many people experience these behaviors occasionally, about 1–5% of individuals develop ongoing patterns that are hard to control and cause significant distress or health problems [66]. Studies report wide differences in how common BFRBs are, with some behaviors affecting a large portion of the population at

different points in life [67]. BFRBs often occur alongside other mental health challenges and can reduce quality of life due to visible physical effects, feelings of shame or anxiety, and social withdrawal [68].

The growing recognition of the importance of monitoring BFRBs has already led to the development of several assessment approaches, which rely primarily on retrospective self-report questionnaires and clinician interviews [64, 65, 69–73]. In clinical studies, treatment response is typically evaluated using clinician-rated and self-reported severity scales for behaviors such as hair pulling and skin picking, rather than through direct behavioral measurement [74]. More direct monitoring approaches, particularly video-based methods, are intrusive, raise privacy concerns, and require impractical form factors, making them unsuitable for longitudinal, free-living use [70, 75]. As a result, despite strong clinical evidence that reductions in repetitive behaviors reflect meaningful improvement, there is currently no scalable, privacy-preserving system capable of continuously monitoring BFRBs in naturalistic settings to objectively quantify symptom change over time highlights a pressing unmet gap.

BFRBs are expressed through a small set of repetitive motor behaviors, many of which are dominated by hand gestures. The most common BFRB subtypes including trichotillomania (hair pulling), excoriation (skin picking), serve as the primary observable biomarkers used to assess BFRB severity. Their strong reliance on hand gestures makes the most prevalent BFRB behaviors particularly well suited for monitoring using wrist- or hand-worn wearable devices in everyday, real-world settings.

Despite recent progress in wearable-based activity and gesture recognition, accurately detecting BFRB-related hand gestures in free-living settings remains challenging. Wearable sensor signals are inherently noisy, multimodal, and temporally complex, with large intra- and inter-subject variability arising from differences in movement style, context, and duration. Moreover, BFRB gestures are often subtle, repetitive, and temporally extended, making them difficult to distinguish from benign daily activities using short-window or static feature representations alone.

Capturing the temporal structure of BFRB-related hand gestures is critical, as these behaviors

unfold causally over time and are embedded within long, continuous streams of daily activity. Convolutional models such as Temporal Convolutional Networks (TCNs) have demonstrated strong performance in multivariate time-series analysis by extracting hierarchical temporal features through dilated convolutions, large receptive fields, and an inherently causal structure that respects temporal ordering. However, while TCNs are effective at modeling local and mid-range temporal dynamics, they lack an explicit mechanism for capturing long-range temporal dependencies and global contextual relationships across extended sequences.

Transformer-based architectures, particularly BERT-style self-attention models [76, 77], address this limitation by enabling global temporal reasoning through self-attention, allowing the model to learn rich contextual interactions across an entire sequence. Prior work [78, 79] has demonstrated that Transformer-based models can outperform recurrent and convolutional approaches for time-series and wearable sensing tasks by effectively modeling long-range temporal dependencies. However, classical Transformer formulations rely on absolute positional encodings [76], typically implemented using sinusoidal functions, which assume fixed sequence lengths and can limit generalization to variable-duration or longer temporal patterns. This limitation becomes particularly significant in longitudinal BFRB monitoring, where gesture durations, repetition rates, and temporal structures can vary substantially across individuals and recording sessions.

To address these challenges, we adopt Rotary Positional Embeddings (RoPE) as an alternative to absolute positional encoding within the BERT-based temporal modeling stage. RoPE encodes relative positional information directly into the self-attention mechanism, enabling the model to preserve temporal ordering while improving extrapolation to longer or variable-length sequences. This property is especially well suited for wearable BFRB monitoring, where continuous sensing and free-living data collection demand robust temporal generalization beyond fixed observation windows. By combining TCN-based hierarchical and causal feature extraction with RoPE-enhanced Transformer modeling, the proposed framework jointly captures fine-grained local dynamics and long-range temporal context, enabling more reliable discrimination of repetitive BFRB gestures

from visually similar everyday hand movements.

3.1.2 Contribution

Building on the above motivations and challenges, the main contributions of this work are summarized as follows:

- We introduce a privacy-preserving wrist-worn sensing framework for detecting body-focused repetitive behavior (BFRB) gestures and distinguishing them from non-BFRB hand movements using multimodal wearable sensor data.
- We propose a hybrid TCN–Transformer architecture that combines hierarchical temporal feature extraction with long-range contextual reasoning through a BERT-style self-attention model.
- We integrate Rotary Positional Embeddings (RoPE) into the Transformer component to improve temporal generalization for variable-length and long-duration gesture sequences.

3.2 Related Work

BFRBs are among the most common yet under-recognized mental health conditions encountered in clinical and community settings [62, 63]. According to the authors [64] main motives for performing BFRBs were release of stress (84.7%), boredom (51.5%), and gratification/pleasure (34.7%). Approximately one third of the sample were unable to provide a clear motive. The majority were ambivalent about their behavior. Participants rarely engaged in cutting; 16.4% performed a BFRB on someone else's body or wanted to do so. OCD was self-reported by only 7.5% of the participants. Prevalence rates for BFRBs as a broad category vary across studies, ranging from 0.5% to 4.4% of the population, however, the rate of occurrence is actually much higher because

many people don't reach out to a healthcare provider for help with these conditions [67]. Where another study showed: Skin Picking: 2% to 20%, Nail Biting: 6% to 34%, Lip/Cheek Biting: 6% to 42% [65]. Up to 60% of the population may experience times in their life when they may get caught up in cycles of picking, biting and pulling behaviors. Approximately 1%-5% of people suffer with body-focused repetitive behaviors, which are recurring patterns of behavior that can be difficult to control. These behaviors often result in significant distress and may lead to serious health consequences [66]. BFRBs often coexist with other mental health challenges as well, creating a cycle that can exacerbate both the behaviors and their consequences and impact quality of life [68]. For instance:

- The visible effects of BFRBs, such as hair loss or skin lesions, may lead to social anxiety, embarrassment, or withdrawal.
- Feelings of shame or guilt about the behaviors can worsen stress and anxiety, further fueling the cycle of BFRBs.
- Over time, this emotional toll can contribute to conditions like depression or generalized anxiety disorder, making it even more challenging to break free from the behaviors.

High rates of comorbidity with anxiety disorders, depression, and obsessive–compulsive disorder further exacerbate disease burden and complicate diagnosis, as BFRBs are frequently overshadowed by co-occurring psychiatric conditions rather than addressed as primary treatment targets [80].

3.2.1 Monitoring and Assessment of BFRBs

Despite growing clinical recognition of the importance of monitoring BFRBs, existing assessment approaches remain fundamentally limited. Current clinical practice relies primarily on retrospective self-report questionnaires and clinician interviews [64, 65, 67, 69–73, 81–85], which are inherently subjective and fail to capture the automatic, episodic, and context-dependent nature of BFRBs as

they occur in daily life [86]. These methods are further compromised by recall bias and limited temporal resolution, making them ill-suited for continuous symptom tracking.

In clinical trials, treatment efficacy is typically inferred from reductions in BFRB severity using standardized clinician-rated and self-reported scales [72, 80, 81, 85, 87, 88]. While these findings demonstrate that reductions in repetitive behaviors reflect meaningful clinical improvement, they do not provide objective, fine-grained evidence of how or when these behaviors change in naturalistic settings.

To address these shortcomings, several technology-assisted monitoring approaches have been explored, including methods based on video, images, or manual observation [69, 70, 72, 89]. However, such approaches are fundamentally incompatible with longitudinal, real-world deployment. Their intrusive nature, privacy risks, high user burden, and impractical form factors severely limit feasibility for free-living monitoring, particularly in sensitive mental health contexts.

3.2.2 Wearable Sensing for Hand Gesture and Behavioral Monitoring

More recently, wearable sensing approaches have been proposed to enable objective, on-body monitoring of behaviors relevant to BFRBs without relying on visual recordings. For example, Son et al [90] introduced a wrist-worn device that combines inertial, proximity, and thermal sensors to estimate hand position relative to the head, motivated by the need to monitor BFRBs such as hair pulling and skin picking. While their results demonstrate that thermal sensing can significantly improve discrimination between different head locations in controlled settings, the study focuses on simulated behaviors and short-term recordings, and does not address continuous, free-living monitoring or the differentiation of BFRBs from everyday activities. As such, despite representing an important step toward privacy-preserving behavioral sensing, these approaches remain insufficient for scalable, longitudinal assessment in naturalistic environments.

Although evidence-based behavioral interventions such as habit reversal training (HRT) demon-

strate strong efficacy in reducing BFRB symptoms, their clinical impact is constrained by limited accessibility, insufficient provider training and awareness, and continued reliance on intermittent, retrospective symptom assessment [89,91,92]. In response to these barriers, commercially available wearable devices that provide real-time vibration feedback such as HabitAware Keen [93], Slightly Robot [94], and FaceTouch Aware [95] have emerged as adjunctive tools aimed at increasing user awareness and interrupting BFRBs in daily life. A recent large-scale manual content analysis of consumer reviews reported high rates of perceived symptomatic improvement and increased awareness among users of these devices [96]. However, the same analysis revealed substantial limitations, including frequent false-positive detections, hardware and software reliability issues, limited behavioral specificity, and a lack of meaningful longitudinal analytics. As a result, these systems function primarily as awareness aids rather than objective, continuous monitoring solutions capable of quantifying behavior dynamics over time.

Unlike commercially available wearable devices that rely on heuristic vibration feedback, recent research efforts have investigated data-driven approaches for automatic BFRB detection. The authors introduced a smartwatch-based system for real-time nail-biting detection using a multi-stage CNN pipeline designed to reduce false positives and optimize power efficiency [97]. Although this approach improves detection accuracy and preserves user privacy by avoiding audio or visual recordings, it is restricted to a single behavior and does not support comprehensive, longitudinal assessment of BFRBs across daily life. As such, it highlights both the promise and current limitations of wearable sensing for objective BFRB monitoring.

3.2.3 Temporal Modeling for Wearable Time-Series Data in BFRBs

Early technology-assisted approaches for monitoring BFRBs relied primarily on rule-based and heuristic temporal logic applied to short windows of wearable sensor data. Commercial devices such as HabitAware Keen [93], Slightly Robot [94], and FaceTouch Aware [95] implement fixed thresholds on motion magnitude, hand-to-head proximity, or posture duration to trigger vibrotactile

feedback, without learning temporal representations of behavior. While these systems improve user awareness, they treat BFRBs as isolated events and lack adaptability to individual behavior or context. A recent large-scale manual content analysis of consumer reviews reported frequent false-positive detections, limited behavioral specificity, and reliability issues, highlighting that such heuristic approaches are poorly suited for continuous, free-living monitoring [96].

More recent academic work has begun applying deep learning models to BFRB-related wearable data, primarily using one-dimensional CNNs to learn short-term temporal patterns directly from inertial signals. For example, Alesmaeil et al. proposed a smartwatch-based nail-biting detection system employing a multi-stage CNN pipeline to reduce false positives while preserving on-device efficiency [97]. Although CNN-based models outperform rule-based and classical machine learning approaches, their fixed receptive fields limit their ability to capture long-range temporal dependencies such as gradual urge buildup, repeated suppression attempts, or habitual behavioral cycles. Recent study further extend CNN-based feature extraction with recurrent architectures such as long short-term memory (LSTM) networks to capture short-range temporal dependencies within segmented windows. However, these models remain fundamentally constrained by windowed processing, limited temporal receptive fields, and challenges in scaling to long-duration, noisy free-living recordings [98].

On the other hand TCNs offer a compelling yet unexplored alternative for BFRB monitoring. By combining causal and dilated convolutions, TCNs can model long-duration temporal dependencies while avoiding future information leakage, making them well suited for real-time, longitudinal wearable sensing [99, 100]. Compared to recurrent models, TCNs provide stable gradients, parallel computation, and improved efficiency on resource-constrained platforms. Despite these advantages, no published BFRB study has explicitly employed TCNs or other long-horizon temporal models, underscoring a critical methodological gap. This limitation motivates the exploration of hierarchical temporal modeling frameworks capable of capturing the evolving, habitual, and context-dependent dynamics that define BFRBs in free-living settings.

3.2.4 Transformer Models for Time-Series and Sensor Data

Transformer architectures, originally introduced for sequence modeling in natural language processing [76], have been increasingly adopted for time-series and wearable sensor data due to their ability to model long-range temporal dependencies via self-attention. Unlike CNNs and TCNs, which rely on fixed or hierarchically expanding receptive fields, Transformers attend globally across entire sequences, enabling flexible modeling of long-duration temporal relationships. Early work adapted positional encodings to preserve temporal order in sensor streams [101, 102], demonstrating improved representation learning for multivariate time-series data.

Building on this paradigm, BERT-style self-attention models treat time-series segments as tokenized sequences and learn contextualized embeddings through self-supervised pretraining [77, 102]. These approaches have shown strong performance in wearable-based activity and gesture recognition, particularly in free-living and long-duration settings, where global temporal context and cross-window dependencies are critical.

Despite their strengths, Transformer models remain computationally expensive due to the quadratic complexity of self-attention and often require careful positional encoding design to generalize across varying sequence lengths [103, 104]. These limitations motivate ongoing research into efficient attention mechanisms and hybrid temporal architectures for scalable, longitudinal wearable sensing.

3.2.5 Positional Encoding Strategies: Absolute vs. Relative

Transformer architectures rely on positional encoding mechanisms to preserve ordering information that is otherwise absent from self-attention. The original BERT formulation employs absolute positional embeddings, which assign each token a fixed position index that is added to its input representation [77]. While effective for fixed-length textual inputs, absolute positional encodings implicitly assume a bounded and stationary sequence length, limiting their ability to generalize to

longer or variable-duration sequences encountered at inference time.

This limitation is particularly problematic for wearable time-series data, where gesture durations, repetition frequency, and temporal structure vary substantially across individuals, sessions, and real-world contexts as shown in Table 3.1. Prior work has shown that absolute positional embeddings exhibit poor extrapolation beyond the sequence lengths observed during training, motivating the exploration of relative and rotation-based alternatives [104, 105].

RoPE were introduced to encode positional information directly within the self-attention mechanism by applying position-dependent rotations to query and key vectors [106]. Unlike absolute embeddings, RoPE enables attention scores to depend on relative token offsets while preserving global position information, allowing Transformers to generalize more effectively to longer and variable-length sequences at inference time, including in post-training settings. Subsequent studies have demonstrated that RoPE improves length extrapolation, stabilizes attention patterns, and enhances performance in settings requiring long-range temporal reasoning [105, 107].

Although RoPE was originally proposed for language modeling, recent analyses have shown that replacing absolute positional embeddings with RoPE in encoder-style architectures, including BERT variants, can mitigate positional bias and improve robustness on sequence labeling and structured prediction tasks [108]. These properties are particularly relevant for longitudinal wearable sensing, where continuous data streams do not admit a natural fixed positional index and where meaningful temporal patterns may span hundreds or thousands of timesteps.

Motivated by these observations, we adopt RoPE in place of absolute sinusoidal positional encoding within the BERT-based temporal modeling stage of our framework. This design choice enables the model to preserve temporal ordering while improving generalization across variable-duration BFRB gesture sequences.

3.3 Problem Formulation and Dataset

3.3.1 Behavioral Time-Series Representation

We formulate BFRB-related hand gesture recognition as a supervised temporal sequence modeling problem using multimodal wrist-worn sensor data collected in free-living environments. Given a time series $\mathbf{X} = \{x_1, \ldots, x_T\}$, where each observation $x_t \in \mathbb{R}^C$ represents synchronized inertial sensor measurements across C channels and the sequence length T varies across instances, the objective is to predict a gesture label $y \in \mathscr{Y}$ corresponding to BFRB-related or non-BFRB hand movements. While gesture variability exists in both controlled and naturalistic settings, free-living data lacks explicit temporal boundaries and exhibits unconstrained variations in gesture duration, repetition, and context within long streams of daily activity.

This problem formulation introduces several key challenges: (1) noisy and heterogeneous multimodal sensor signals, (2) high intra- and inter-subject variability in movement patterns, (3) long-range temporal dependencies arising from extended and repetitive gestures, and (4) variable sequence lengths that limit the effectiveness of models relying on fixed positional assumptions. Addressing these challenges requires a temporal modeling framework capable of jointly capturing fine-grained local dynamics and long-range contextual relationships while generalizing across variable-duration sequences.

To this end, we propose a hybrid architecture that combines TCN for causal and hierarchical feature extraction with a Transformer-based self-attention model enhanced using RoPE. This design enables robust temporal reasoning over extended sequences while preserving ordering information and improving extrapolation to longer or unseen sequence lengths, which is essential for free-living BFRB monitoring.

3.3.2 Dataset Overview

The dataset used in this study was developed by the Child Mind Institute in collaboration with the Healthy Brain Network and is designed to support research on wearable sensing for BFRBs [109]. It comprises a total of 574,945 labeled gesture samples collected from 81 unique participants, corresponding to approximately 1.2 GB of multimodal sensor data. Data were acquired using the Helios wrist-worn sensing platform, which integrates inertial, thermal, and proximity sensing modalities detailed in Table 3.1.

Table 3.1: Comprehensive summary of the gesture dataset, including the number of subjects and samples, gesture category distribution, and demographic characteristics of participants. The table also reports detailed sequence length statistics, highlighting substantial variability in gesture durations across samples, as well as handedness and physical measurement information.

Dataset Characteristic	**Value**
Total Unique Subjects	81
Total Gesture Samples	574,945
Gesture Information	
Unique Gesture Types	18
Most Common Gesture	Text on phone
Least Common Gesture	Pinch knee/leg skin
Gesture Length Statistics (Sequence Length)	
Minimum Length	29 samples
Maximum Length	700 samples
Mean Length	70.5 ± 35.4 samples
Median Length	59.0 samples
Length Range	29 – 700 samples
Demographics – Age	
Age Range	10 – 53 years
Mean Age	21.8 ± 10.3 years
Median Age	22.0 years
Demographics – Age Group	
Children ($<$13 years)	18 (22.2%)
Teens (13–18 years)	21 (25.9%)
Young Adults (19–30 years)	26 (32.1%)
Adults (31–50 years)	14 (17.3%)
Seniors ($>$50 years)	2 (2.5%)
Demographics – Gender	
Male	50 (61.7%)
Female	31 (38.3%)
Demographics – Handedness	
Right-handed	71 (87.7%)
Left-handed	10 (12.3%)
Physical Measurements	
Mean Height	168.0 ± 10.6 cm
Height Range	135.0 – 190.5 cm
Mean Shoulder-to-Wrist	51.6 ± 4.9 cm
Mean Elbow-to-Wrist	25.5 ± 3.0 cm

3.3.3 Participants and Demographics

The participant cohort includes 81 individuals (50 males and 31 females) with ages ranging from 10 to 53 years (mean: 21.8 $\pm$ 10.3 years; median: 22.0 years). The cohort consists of both adults (51.9%, $n = 42$) and children (48.1%, $n = 39$). Most participants reported right-handed dominance (87.7%), with the remainder reporting left-handed or ambidextrous use.

Physical measurements were recorded for each participant, including height (mean: 168.0 $\pm$ 10.6 cm; range: 135.0–190.5 cm), shoulder-to-wrist length (mean: 51.6 $\pm$ 4.9 cm), and elbow-to-wrist length (mean: 25.5 $\pm$ 3.0 cm). Strong correlations were observed between height and shoulder-to-wrist length ($r = 0.73$), as well as between shoulder-to-wrist and elbow-to-wrist length ($r = 0.67$), supporting their use for normalization in gesture analysis.

3.3.4 Gesture Set and Class Distribution

The dataset contains 18 distinct gesture classes, consisting of 8 BFRB-like target behaviors and 10 non-BFRB comparison gestures. Target gestures include multiple forms of hair pulling, skin pinching, and scratching, while non-target gestures include everyday activities such as phone texting, drinking, and manipulating glasses.

Class frequencies are imbalanced, with *Text on phone* (58,462 samples; 10.2%) and *Neck–scratch* (56,619 samples; 9.8%) being the most frequent gestures, and *Pinch knee/leg skin* (9,844 samples; 1.7%) being the least frequent. Each participant performed between 5 and 18 distinct gesture types, yielding a total of 8,151 unique gesture sequences. On average, each participant contributed 7,098.1 samples (median: 6,954.0) as shown in Fig. 3.1.

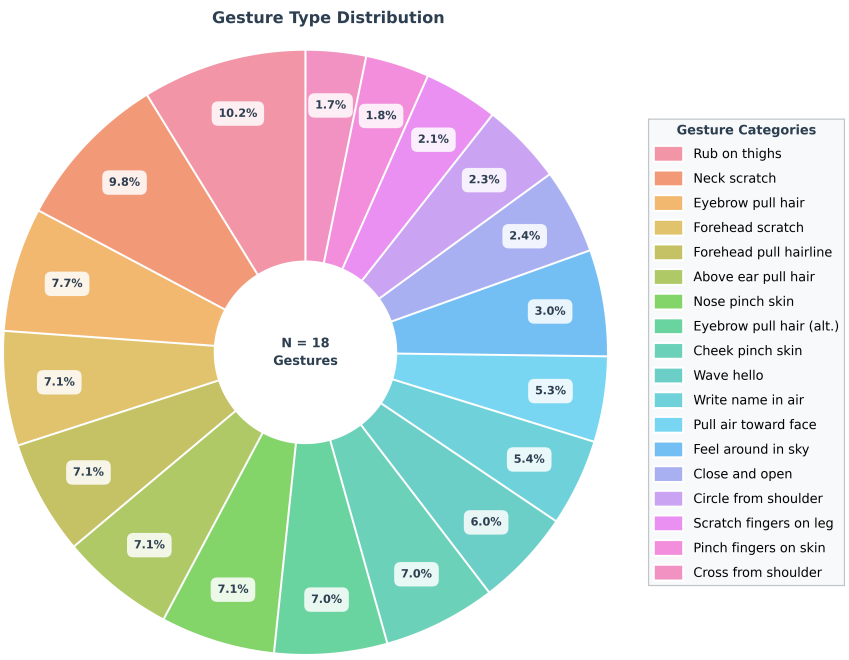

Figure 3.1: Distribution of gesture categories used in the study. The dataset contains 18 gesture types, including BFRB-like behaviors and non-BFRB control gestures, illustrating the diversity of hand-to-face and everyday movements used for model training and evaluation.

3.3.5 Sensing Modalities and Feature Representation

Data were recorded using the Helios wrist-worn device, which integrates multiple sensing modalities. Inertial data include 3-axis accelerometer measurements (acc_x, acc_y, acc_z) and quaternion-based rotational representations (rot_w, rot_x, rot_y, rot_z). Thermal sensing is provided by five thermopile sensors (thm_1–thm_5), which measure infrared radiation corresponding to skin proximity, with typical readings ranging from 24–32°C.

Proximity sensing is achieved via five time-of-flight (ToF) sensors, each producing 64 distance measurements per timestep, resulting in 320 ToF features. Distance values typically range from 50–250 mm, with out-of-range measurements encoded as -1. Together, these modalities provide complementary kinematic, spatial, and proximity information for differentiating gestures occurring at different body locations.

The complete feature representation consists of 341 columns, including sensor measurements and

structured metadata.

3.3.6 Temporal Structure and Annotation Schema

Gesture data were collected under standardized protocols across four body orientations: sitting upright, sitting while leaning forward, lying supine, and lying on the side. Each gesture sequence is temporally segmented into three phases: *Transition*, corresponding to hand movement toward the target location; *Pause*, representing brief stillness; and *Gesture*, during which the behavior is performed.

Each sample includes hierarchical metadata fields such as *sequence_type* (target or non-target), *sequence_id*, *sequence_counter*, *subject*, *orientation*, *behavior*, *phase*, and *gesture*. This structured annotation enables sequence-level and phase-aware modeling, supporting approaches that leverage temporal progression and contextual information for gesture recognition.

3.4 Methodology

This section presents a temporal modeling framework for recognizing BFRB-related hand gestures from multimodal wearable sensor data, addressing challenges of variable-length sequences, noisy free-living signals, and long-range temporal dependencies. As illustrated in Fig. 3.2, the framework employs a hierarchical architecture: parallel TCN branches with Squeeze-and-Excitation attention extract multi-scale features from accelerometer and gyroscope, thermal, and time-of-flight sensors; these features are fused and processed by a BERT-based Transformer encoder with RoPE to capture global context while enabling extrapolation to sequences longer than the training length. An ensemble of 10 independently trained models uses soft voting to improve robustness. This design jointly models local motion dynamics and long-range temporal patterns while maintaining adaptability to variable-length gesture sequences.

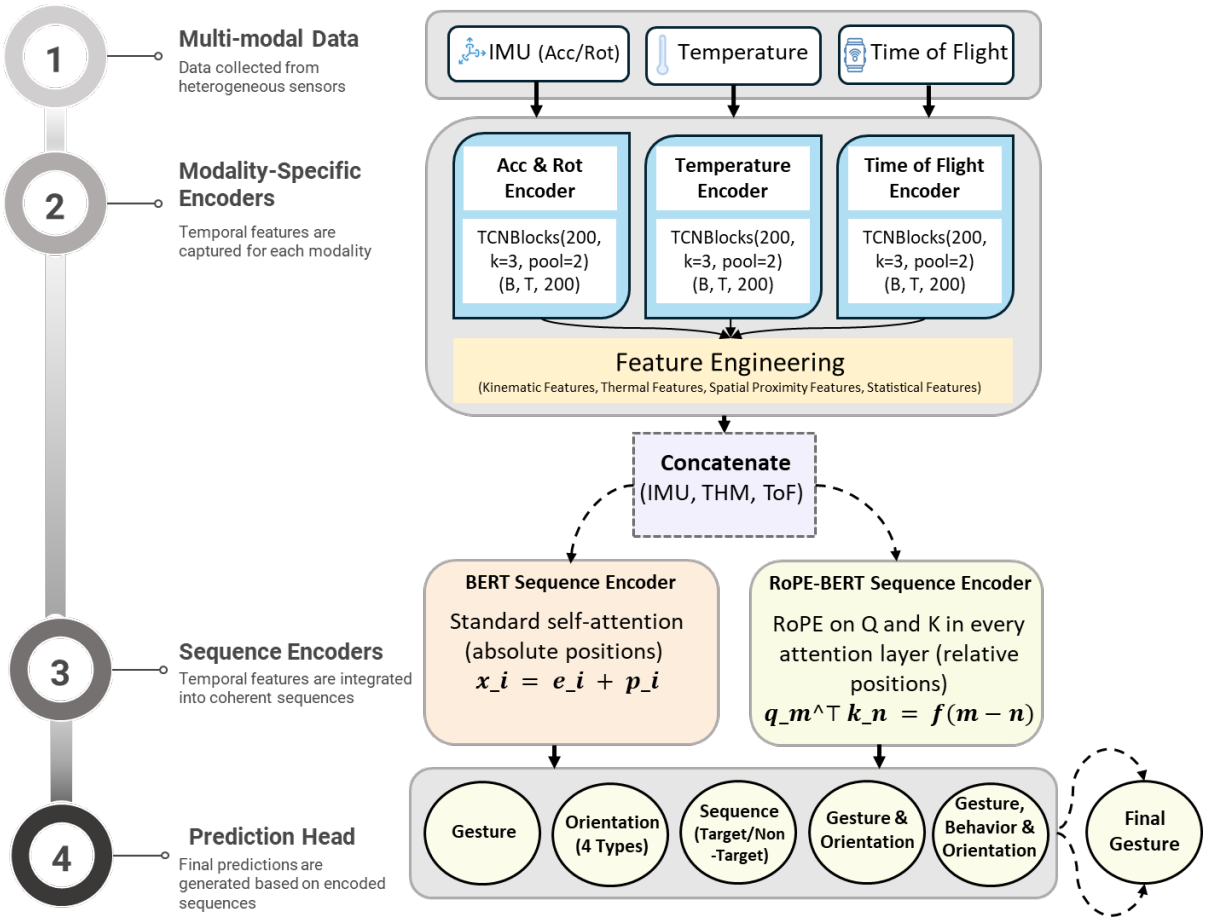

Figure 3.2: Multimodal gesture recognition pipeline where sensor inputs (IMU, temperature, and ToF) are processed using modality-specific TCN encoders, followed by feature fusion and sequence modeling with either a BERT or RoPE-BERT transformer to produce the final gesture prediction.

3.4.1 Data Acquisition and Preprocessing

The methodology begins with comprehensive data acquisition from multi-modal wearable sensors designed to capture hand-to-body interaction gestures. Each sequence is identified by a unique sequence identifier that corresponds to a single complete gesture instance performed by a participant, ensuring temporal coherence and proper grouping of sensor readings across the duration of the gesture execution.

The raw sensor stream includes three primary modalities: IMU branch that combines linear acceleration measurements (from the accelerometer) and angular velocity measurements (from the

gyroscope) in three-dimensional space, temperature sensors providing thermal signatures of hand proximity, and ToF sensors generating distance measurements across spatial grids. Sequences exhibit natural variability in duration, ranging from approximately 29 to 700 temporal samples, with an average length of 72.44 samples, reflecting the inherent diversity in gesture execution patterns across different gesture classes and individual participants.

3.4.2 Sequence Standardization and Context Length Management

To accommodate the fixed-length requirements of deep learning architectures while preserving the temporal characteristics of variable-length gestures, a standardization procedure is implemented. Sequences shorter than the target context length of 300 timesteps are zero-padded at the end, preserving the initial gesture onset and maintaining temporal alignment. Conversely, sequences exceeding 300 timesteps undergo truncation, retaining the final 300 frames to capture the critical gesture completion phase, which typically contains the most discriminative motion patterns for classification tasks.

This approach ensures consistent input dimensionality across all sequences while prioritizing the preservation of gesture-specific temporal dynamics. The masking mechanism distinguishes valid timesteps from padding, enabling the model to appropriately weight information during both training and inference phases. The choice of 300 timesteps as the context length balances the need to capture complete gesture patterns with computational efficiency, accommodating the majority of gesture instances while providing sufficient temporal resolution for fine-grained motion analysis.

3.4.3 Feature Engineering and Sensor Modality Processing

The preprocessing pipeline handles each sensor modality independently before fusion, recognizing that different sensor types exhibit distinct temporal and spectral characteristics. For the IMU branch, linear acceleration components in three dimensions are extracted after gravity removal, which

transforms accelerometer readings from the sensor frame to the world frame using quaternion-based rotations. Derived features include acceleration magnitude ($\sqrt{a_x^2 + a_y^2 + a_z^2}$), acceleration magnitude jerk (temporal derivative), and angular distance, which quantifies cumulative orientation changes between consecutive quaternion rotations. Angular velocity measurements provide rotational dynamics in three dimensions. These derived features capture both directional motion and intensity variations, which are particularly relevant for distinguishing between subtle repetitive gestures and transient movements.

Temperature sensor readings from five discrete thermal sensors are processed to capture proximity-based thermal signatures. These measurements undergo normalization procedures using mean and standard deviation to account for ambient temperature variations and individual physiological differences, ensuring consistent representation across participants and environmental conditions.

The ToF sensor array, comprising five sensors each generating 64-pixel spatial measurements, provides rich proximity and spatial interaction information. The preprocessing pipeline extracts both raw distance values (normalized by dividing by 255) and statistical features computed over pixel regions. For each of the 16 regions per sensor (dividing the 8×8 grid), mean, standard deviation, minimum, and maximum values are computed, resulting in 4 global statistics plus 64 region-specific statistics (16 regions × 4 statistics) per sensor. Combined with the 64 raw pixel values, each ToF sensor contributes 68 statistical features plus 64 raw features, totaling 660 features across all five sensors.

3.4.4 Temporal Convolutional Network Architecture with Squeeze and Excitation

The proposed framework employs TCNs to model multi-scale temporal dependencies in multimodal wearable sensor data. Non-causal symmetric dilated convolutions are used with padding computed as $(k-1) \times d/2$, allowing each temporal window to exploit both past and future context. This

design is suitable for window-based gesture recognition, where complete gesture sequences are available at inference time, in contrast to online streaming scenarios that require causal modeling.

Each modality is processed by an independent TCN branch composed of stacked *TemporalBlocks* with exponentially increasing dilation rates $[1,2,4]$. Each TemporalBlock consists of two one-dimensional convolutional layers (kernel size $= 3$), each followed by batch normalization, ReLU activation, and dropout. Residual connections, implemented via identity mappings or 1×1 convolutions when channel dimensions differ, are applied before the final activation to stabilize optimization and preserve low-level temporal features. With kernel size 3 and dilation rates $[1,2,4]$, the effective receptive field spans approximately 21 timesteps, enabling the network to capture both fine-grained motion transitions and mid-range temporal patterns.

To enhance channel-wise feature discriminability, Squeeze-and-Excitation (SE) attention is integrated after the dual-convolution structure in each TemporalBlock. Global average pooling over the temporal dimension produces channel descriptors, which are passed through a two-layer bottleneck MLP with a reduction ratio of 8 and sigmoid activation to generate channel-wise recalibration weights. These weights are applied via channel-wise scaling, allowing the network to emphasize informative sensor features with negligible computational overhead.

The fusion of multimodal features is performed through concatenation along the feature (channel) dimension after each sensor branch has been independently processed by its respective TCN. The IMU, temperature, and ToF branch outputs, each of shape $(B, L, 200)$, are concatenated to form a unified representation of shape $(B, L, 600)$. This fused tensor integrates motion dynamics, thermal signatures, and proximity information while preserving temporal alignment across modalities, enabling the transformer-based sequence model to jointly reason over temporally synchronized multimodal features.

3.4.5 Transformer-Based Sequence Modeling with Rotary Position Embedding

The fused multimodal feature sequence is modeled using a BERT-style transformer encoder. A learnable classification (CLS) token is prepended to the input sequence and serves as a global representation that aggregates temporal information through self-attention. The final hidden state of the CLS token is used for downstream gesture classification.

The transformer consists of 8 encoder layers with a hidden dimension of 600, 10 attention heads, and a feed-forward dimension of 2,400. Each layer follows the standard transformer structure with multi-head self-attention, position-wise feed-forward networks, residual connections, and layer normalization, enabling modeling of long-range temporal dependencies and cross-modal interactions.

To enable variable-length sequence modeling and context-length extrapolation, RoPE is used in place of absolute positional embeddings. RoPE injects positional information by applying position-dependent rotations to the query and key vectors in each attention head, such that attention scores depend on relative positional offsets rather than absolute indices.

For each attention head with dimension d_h, the query and key vectors are split into pairs of features. For position p and pair index i, the rotation angle is defined as

$$\theta_i = \text{base}^{-2i/d_h}, \qquad \alpha_{p,i} = p \cdot \theta_i, \tag{3.1}$$

where the base is set to 10,000. The rotation is applied to each feature pair (x_{2i}, x_{2i+1}) as

$$\begin{bmatrix} x'_{2i} \\ x'_{2i+1} \end{bmatrix} = \begin{bmatrix} \cos(\alpha_{p,i}) & -\sin(\alpha_{p,i}) \\ \sin(\alpha_{p,i}) & \cos(\alpha_{p,i}) \end{bmatrix} \begin{bmatrix} x_{2i} \\ x_{2i+1} \end{bmatrix}, \tag{3.2}$$

and the same transformation is applied to both query and key vectors. This formulation ensures that attention scores depend only on relative positional differences between tokens, enabling natural generalization to longer sequences.

RoPE is particularly suitable for gesture recognition, where relative temporal relationships between motion phases are more informative than absolute timestep indices. It also supports seamless evaluation on sequences longer than those observed during training, facilitating context-length extrapolation studies.

Following transformer encoding, the CLS token embedding provides a compact sequence-level representation, which is passed to the classification head for final prediction. To enable controlled comparison between absolute and rotary positional encoding schemes, the transformer backbone supports two configurable modes. In the standard setting, the model uses BERT's learned absolute positional embeddings, which are added to the sensor token representations prior to self-attention. In the RoPE setting, absolute positional embeddings are disabled and positional information is instead injected directly into the attention mechanism via rotary transformations of query and key vectors, as described above. The positional encoding strategy is selected at training time and remains fixed throughout each experiment, ensuring a fair and isolated evaluation of the impact of positional encoding on sequence modeling and length extrapolation.

3.4.6 Multi-Task Learning and Classification Heads

The contextualized CLS token embedding $(B, 600)$ is shared across multiple task-specific classification heads, enabling joint optimization of related objectives through multi-task learning. This strategy encourages the model to learn representations that are simultaneously discriminative for multiple tasks, improving generalization via shared supervision and implicit regularization.

The primary gesture classification head maps the CLS embedding to 18 gesture classes using a linear layer. Auxiliary heads are introduced for device orientation (4 classes) and sequence type (binary),

each implemented using a single linear projection. In addition, two joint classification heads predict combined labels: (i) gesture $\times$ orientation (72 classes) and (ii) gesture $\times$ orientation $\times$ behavior (144 classes), implemented using lightweight multi-layer perceptrons with batch normalization and dropout.

Training is performed using a weighted sum of cross-entropy losses over all active tasks:

$$\mathscr{L}_{\text{total}} = \sum_{t} w_t \mathscr{L}_t, \tag{3.3}$$

with equal weights assigned to all enabled tasks. This formulation promotes balanced learning across objectives while allowing flexible task inclusion through loss weighting.

3.4.7 Data Augmentation

To improve robustness and generalization, data augmentation is applied during training only. The augmentation pipeline simulates realistic variations in gesture execution and sensor measurements, including partial gesture capture and sensor noise.

Temporal dropout is applied at the beginning and end of sequences to simulate delayed onset and early termination of gestures. Additionally, Gaussian noise is injected into IMU features to model sensor noise and minor placement variations. A phase-aware sequence mixing strategy further augments transition segments by combining compatible gesture sequences with matching labels, increasing diversity in gesture initiation patterns while preserving class consistency.

Augmentations are applied stochastically and sequentially prior to normalization and padding, ensuring diverse training samples without affecting validation or test distributions.

3.4.8 Ensemble Learning Framework

The methodology employs an ensemble learning strategy that combines predictions from 10 independently initialized model instances. Each model shares the same architectural configuration but differs in weight initialization, resulting in diverse feature representations and prediction patterns. This diversity is crucial for ensemble performance, as different models may capture complementary aspects of the gesture recognition problem.

Ensemble prediction is performed through probability averaging, where probability distributions from all models are averaged element-wise before applying argmax to obtain the final prediction.

3.5 Experimental Setup and Evaluation

3.5.1 Cross-Validation Protocol

Evaluation follows a subject-wise cross-validation scheme to prevent identity leakage across splits. All sequences from a given participant are assigned exclusively to either training, validation, or test sets. For each fold, subjects are randomly shuffled and partitioned into 80% training, 10% validation, and 10% testing subsets.

Split assignments are saved and reused across experiments to ensure identical data partitions when comparing different model variants, including absolute and rotary positional encoding configurations. This protocol provides a realistic assessment of generalization to unseen users, which is critical for wearable sensing applications.

3.5.2 Context-Length Extrapolation Evaluation

A distinctive aspect of this methodology is the evaluation of context-length extrapolation capabilities, specifically enabled by the RoPE positional encoding. Models are trained on sequences of 300 timesteps, but evaluated on longer sequences (400, 500, and 600 timesteps) to assess the model's ability to generalize to sequences beyond the training distribution. This evaluation is particularly relevant for real-world scenarios where gesture durations may vary significantly, and the ability to handle longer sequences without retraining represents a significant practical advantage.

The extrapolation evaluation maintains the same preprocessing pipeline, with sequences longer than 300 timesteps truncated to the specified test length while preserving the adaptive pooling operations that ensure consistent feature dimensions. The RoPE encoding naturally accommodates these longer sequences through its relative position encoding mechanism, enabling meaningful inference without architectural modifications or fine-tuning.

3.5.3 Individual Model Performance Analysis

The methodology includes systematic analysis of individual model performance within the ensemble, enabling identification of model diversity and performance characteristics. Each model's accuracy and F1-score are computed independently on the test set, allowing for ranking and comparison. This analysis provides insights into ensemble composition, identifying models that contribute most significantly to ensemble performance and those that may be candidates for removal or further investigation.

Summary statistics across all models, including best-performing models, worst-performing models, average performance, and standard deviation, provide quantitative measures of ensemble diversity and stability. Models exhibiting consistently poor performance across multiple folds may indicate initialization issues or training instabilities, while high-performing models contribute valuable

predictive power to the ensemble.

3.5.4 Training Configuration and Hyperparameters

Models are trained using AdamW optimization with cosine annealing learning rate scheduling and linear warm-up. A batch size of 64 is used for all experiments. The base learning rate is set to 5×10^{-4} for models with absolute positional embeddings and 2×10^{-4} for RoPE-based models to ensure stable optimization.

Gradient clipping is applied to mitigate exploding gradients, with tighter thresholds for RoPE-based models due to the modified attention dynamics. Mixed-precision training is disabled to avoid numerical instability introduced by trigonometric operations in rotary embeddings. All models are trained for 40 epochs on a single GPU with identical data splits and training protocols to ensure fair comparison.

3.5.5 Implementation and Computational Considerations

The proposed framework is implemented in PyTorch 2.x and executed on a workstation equipped with two NVIDIA RTX 4090 GPUs (24 GB each) with full CUDA acceleration enabled for all compute-intensive operations. The implementation leverages PyTorch's CUDA backend and cuDNN-optimized kernels for convolutional layers in the TCN branches, as well as fused CUDA kernels for transformer attention via the scaled dot-product attention SDPA interface.

Data loading and preprocessing are parallelized using multi-worker DataLoaders (32 workers) to mitigate CPU-side bottlenecks introduced by quaternion-based motion processing and region-wise ToF feature aggregation. Page-locked (pinned) host memory is enabled to accelerate CPU–GPU data transfer, and a custom collation function handles padding, truncation, and attention mask generation for variable-length sequences.

Transformer self-attention is computed using CUDA-optimized SDPA kernels, which provide a more efficient implementation than standard eager attention. In benchmark testing on GPU, SDPA achieved an approximately 1.3× speedup over the standard eager attention implementation. Mixed-precision training is disabled for RoPE-based models because rotary embeddings involve trigonometric operations that are more numerically sensitive, so full-precision computation is used to maintain stable training.

The overall model contains approximately 15–20 million trainable parameters, with the majority residing in the transformer encoder. For training sequences of 300 timesteps and batch size 64, GPU memory usage remains within single-GPU limits, while extrapolation evaluation at longer sequence lengths increases memory consumption due to the quadratic complexity of self-attention.

All experiments are conducted under identical software and hardware conditions, with centralized configuration management and automated logging of model checkpoints, predictions, and experimental metadata to ensure reproducibility and fair comparison between absolute and rotary positional encoding variants.

3.6 Results and Analysis

3.6.1 Overall Gesture Recognition Performance

We compare the baseline TCN + BERT model using sinusoidal positional encodings with the proposed TCN + RoPE-BERT architecture at the training context length of 300 timesteps. All results are reported at the sequence level using subject-wise 5-fold cross-validation. Table 3.2 gives accuracy and F1 by fold and context length.

At 300 timesteps, the original BERT ensemble attains a 5-fold mean accuracy of 78.07% and a mean F1 of 77.64%, while RoPE-BERT reaches 80.76% accuracy and 80.55% F1, yielding an average improvement of +2.69 percentage points (accuracy) and +2.91 points (F1). RoPE-BERT

outperforms the baseline on every fold for both metrics. The best single-fold result is in Fold 1: RoPE-BERT reaches 86.00% accuracy and 85.71% F1 versus 84.00% and 83.33% for the baseline, preserving a clear margin even in the most favourable split. Across folds, the baseline accuracy ranges from 74.60% (Fold 2) to 84.00% (Fold 1), and RoPE-BERT from 78.10% (Fold 0) to 86.00% (Fold 1). The largest accuracy gains at 300 timesteps are in Fold 2 (+4.25%) and Fold 0 (+2.61%), with smaller but consistent gains in Folds 3 and 4 (+2.61% and +1.95%). RoPE-BERT thus improves both accuracy and F1 over the sinusoidal baseline at the training context length, with no fold where the baseline wins.

3.6.2 Extrapolation to Longer and Unseen Sequence Lengths

To assess extrapolation, all models were trained with a fixed context of 300 timesteps and evaluated at 400, 500, and 600 timesteps without retraining or fine-tuning. Table 3.2 summarizes the results and reports, for each fold at extrapolated lengths, the change in performance relative to that fold's result at 300 timesteps (italic sub-rows: negative values indicate degradation).

At the training length (300), 5-fold average accuracy is 78.07% for the original BERT ensemble and 80.76% for RoPE-BERT (+2.69%). When the context is extended to 400 timesteps (+100 beyond training), the original BERT average falls to 77.46% while RoPE-BERT reaches 80.84% (+3.38% improvement). At 500 timesteps (+200), the baseline drops to 76.59% and RoPE-BERT to 80.30% (+3.71%). At 600 timesteps (+300), the figures are 76.61% and 79.76% respectively (+3.15%). F1 follows the same pattern: at 300 the means are 77.64% (original) and 80.55% (RoPE); at 400 they are 77.00% and 80.58% (+3.58% RoPE over original); at 500, 76.17% and 80.00% (+3.83%); at 600, 76.12% and 79.51% (+3.39%).

Two patterns are evident also can be seen in Fig. 3.3. First, the original BERT ensemble degrades as context length increases beyond training. In accuracy, the 5-fold average drops by 0.61 points from 300 to 400, by 1.48 points to 500, and by 1.46 points to 600 (relative to 300). In F1, the

corresponding drops are 0.64, 1.47, and 1.52 points. The per-fold sub-rows in the table show that this degradation is consistent across folds: for example, in Fold 0 the original model loses 1.96 points in accuracy at 400, 3.92 at 500, and 3.59 at 600, whereas RoPE-BERT in the same fold loses 0.65 points at 500 and 600 and actually gains +0.66 at 400. Second, RoPE-BERT is more stable. On average, from 300 to 400 it gains +0.08% accuracy and +0.03% F1; from 300 to 500 it loses 0.46% accuracy and 0.55% F1; from 300 to 600 it loses 1.00% accuracy and 1.04% F1 substantially less than the baseline in each case. In some folds RoPE-BERT improves at longer context (e.g., Fold 0 and Fold 2 at 400; Fold 2 at 500 with +0.12% accuracy vs. 300; Fold 4 at 600 with +0.07% F1 vs. 300). The accuracy gap in favour of RoPE grows with extrapolation from +2.69% at 300 to +3.38% at 400 and +3.71% at 500, then remains strong at 600 (+3.15%). These gains are achieved without architectural changes or re-optimization at longer lengths, indicating that the RoPE-based encoder generalizes beyond the temporal scale seen during training. Encoding relative position in self-attention therefore allows the model to use longer context more effectively than the sinusoidal baseline and to limit degradation when evaluated beyond 300 timesteps a desirable property for free-living BFRB monitoring, where gesture duration and context length vary.

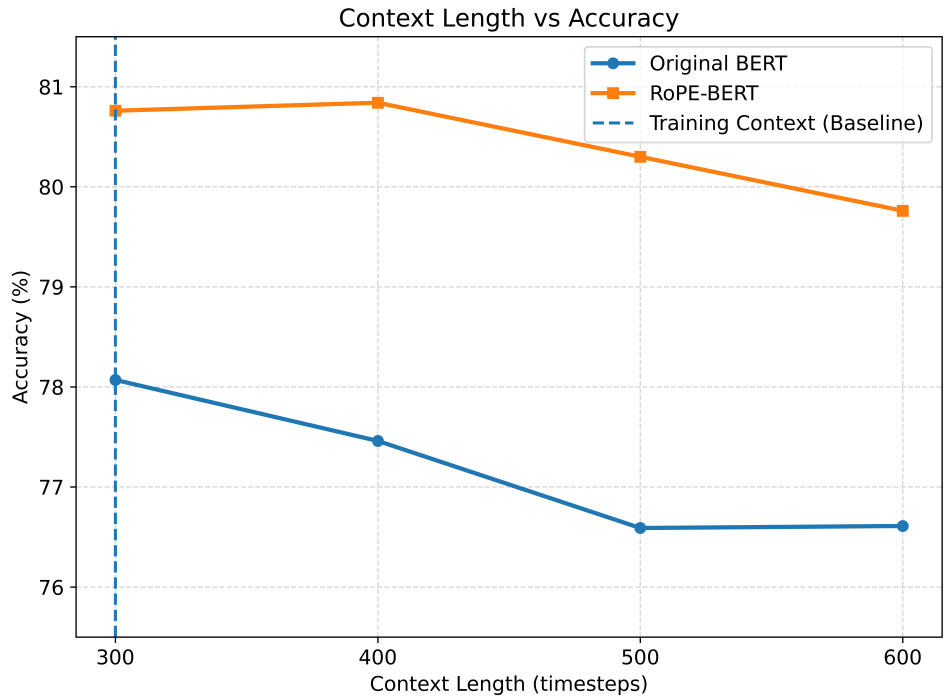

Figure 3.3: Comparison of 5-fold average accuracy across context lengths for the original BERT ensemble and RoPE-BERT. Both models were trained at 300 timesteps and evaluated without retraining at longer unseen lengths (400, 500, and 600). RoPE-BERT maintains better stability and higher accuracy as context length increases.

3.6.3 Impact of Positional Encoding on Temporal Generalization

The observed gap in extrapolation behavior is explained by how each encoding represents position. Sinusoidal encodings fix a vector per absolute index; positions beyond the training length (300) are unseen, so the model degrades when evaluated at longer contexts. RoPE encodes relative distances between tokens, so the same geometry applies at any sequence length. The results are consistent with this: the baseline degrades at 400–600 timesteps while RoPE-BERT remains stable or gains, and the advantage of RoPE grows with extrapolation without retraining. Positional encoding choice is therefore a key determinant of temporal generalization for variable-length gesture recognition.

Table 3.2: Sequence-level accuracy (%) and F1 (%) by fold and context length. Original vs. RoPE-BERT; Δ = RoPE improvement. **Bold** = best per metric. *Italic sub-rows*: change vs. training context (300). *Avg.* = 5-fold average.

		Accuracy (%)			F1 (%)		
Fold	**Context**	Orig.	RoPE	Δ	Orig.	RoPE	Δ
Context length 300 (training)							
0	300	75.49	**78.10**	**+2.61**	74.73	**77.82**	**+3.09**
1	300	84.00	**86.00**	**+2.00**	83.33	**85.71**	**+2.38**
2	300	74.60	**78.85**	**+4.25**	74.21	**78.49**	**+4.28**
3	300	80.07	**82.68**	**+2.61**	79.84	**82.88**	**+3.04**
4	300	76.21	**78.16**	**+1.95**	76.09	**77.84**	**+1.75**
Avg.	300	78.07	**80.76**	**+2.69**	77.64	**80.55**	**+2.91**
Context length 400 (extrapolation, +100)							
0	400	73.53	**78.76**	**+5.23**	72.85	**78.67**	**+5.82**
	vs. 300	−1.96	+0.66		−1.88	+0.85	
1	400	83.67	**86.00**	**+2.33**	82.95	**85.68**	**+2.73**
	vs. 300	−0.33	0.00		−0.38	−0.03	
2	400	74.60	**78.85**	**+4.25**	74.19	**78.53**	**+4.34**
	vs. 300	0.00	0.00		−0.02	+0.04	
3	400	79.74	**82.68**	**+2.94**	79.37	**82.45**	**+3.08**
	vs. 300	−0.33	0.00		−0.47	−0.43	
4	400	75.75	**77.93**	**+2.18**	75.64	**77.57**	**+1.93**
	vs. 300	−0.46	−0.23		−0.45	−0.27	
Avg.	400	77.46	**80.84**	**+3.38**	77.00	**80.58**	**+3.58**
Avg. vs. 300		−0.61	+0.08		−0.64	+0.03	
Context length 500 (extrapolation, +200)							
0	500	71.57	**77.45**	**+5.88**	70.99	**77.31**	**+6.32**
	vs. 300	−3.92	−0.65		−3.74	−0.51	
1	500	82.33	**85.33**	**+3.00**	81.74	**84.95**	**+3.21**
	vs. 300	−1.67	−0.67		−1.59	−0.76	
2	500	73.68	**78.97**	**+5.29**	73.19	**78.65**	**+5.46**
	vs. 300	−0.92	+0.12		−1.02	+0.16	
3	500	79.74	**81.70**	**+1.96**	79.37	**81.41**	**+2.04**
	vs. 300	−0.33	−0.98		−0.47	−1.47	
4	500	75.63	**78.05**	**+2.41**	75.56	**77.70**	**+2.14**
	vs. 300	−0.58	−0.11		−0.53	−0.14	
Avg.	500	76.59	**80.30**	**+3.71**	76.17	**80.00**	**+3.83**
Avg. vs. 300		−1.48	−0.46		−1.47	−0.55	
Context length 600 (extrapolation, +300)							
0	600	71.90	**77.45**	**+5.55**	71.01	**77.35**	**+6.34**
	vs. 300	−3.59	−0.65		−3.72	−0.47	
1	600	82.67	**83.33**	**+0.66**	82.03	**82.98**	**+0.95**
	vs. 300	−1.33	−2.67		−1.30	−2.73	
2	600	73.10	**78.51**	**+5.40**	72.63	**78.22**	**+5.59**
	vs. 300	−1.50	−0.34		−1.58	−0.27	
3	600	79.74	**81.37**	**+1.63**	79.37	**81.10**	**+1.73**
	vs. 300	−0.33	−1.31		−0.47	−1.78	
4	600	75.63	**78.16**	**+2.53**	75.58	**77.91**	**+2.33**
	vs. 300	−0.58	0.00		−0.51	+0.07	
Avg.	600	76.61	**79.76**	**+3.15**	76.12	**79.51**	**+3.39**
Avg. vs. 300		−1.46	−1.00		−1.52	−1.04	

3.7 Discussion

The results support the use of a hybrid TCN–Transformer architecture with RoPE for BFRB-related gesture recognition from wrist-worn sensor data. The consistent gains of RoPE-BERT over the sinusoidal baseline at the training context length, and its stronger performance under extrapolation, align with the motivation that relative positional encoding should better support variable-length and long-duration temporal patterns encountered in free-living settings.

Several factors likely contribute to these outcomes. First, BFRB gestures are repetitive and extend over variable durations; encoding position in a way that generalizes across length RoPE allows the model to attend over longer or shorter windows without being tied to fixed absolute indices. Second, the TCN stage provides hierarchical features that summarize local motion and thermal/proximity cues before the transformer; the transformer then reasons over these summaries with global attention. RoPE in this stage improves how the model uses temporal order when aggregating information across the sequence, which matters when gesture phases (e.g., transition, pause, gesture) have variable length. Third, the subject-wise evaluation protocol ensures that gains are measured on unseen participants, which is relevant for deployment where the system must generalize to new users.

The extrapolation behavior has direct practical implications. In longitudinal BFRB monitoring, gesture duration and the amount of surrounding context available will vary across sessions and individuals. A model that degrades when given longer context (as observed with sinusoidal encoding) would require careful tuning of window length or repeated retraining. By contrast, RoPE-BERT maintains or slightly improves at 400 timesteps and degrades less than the baseline at 500 and 600, so a single trained model can be applied across a range of context lengths without retraining. This supports the use of the same system in settings where recording conditions or annotation practices lead to variable effective sequence lengths.

The present work does not compare directly against commercial BFRB wearables or other deep

learning baselines on the same dataset; such comparisons would strengthen the evidence. Nevertheless, the improvement over a strong TCN + BERT baseline with standard sinusoidal encoding, together with the extrapolation results, indicates that the choice of positional encoding is important when temporal context length is variable. Future work could extend the evaluation to additional datasets, other BFRB subtypes, and longer or shorter context lengths to better characterize the limits of temporal generalization.

3.8 Conclusion

This work presented a privacy-preserving, wrist-worn sensing framework for recognizing BFRB-related hand gestures from multimodal wearable data in free-living settings. The framework combines Temporal Convolutional Networks for hierarchical, causal feature extraction from inertial, thermal, and proximity sensors with a BERT-based Transformer encoder for long-range temporal reasoning. RoPE were incorporated in place of sinusoidal positional encoding to improve robustness to variable-length and extended gesture sequences.

Experiments under subject-wise 5-fold cross-validation showed that the proposed TCN + RoPE-BERT ensemble outperforms the same architecture with sinusoidal encoding at the training context length (300 timesteps), with mean accuracy and F1 gains of about 2.7 and 2.9 percentage points, and with RoPE-BERT ahead on every fold. When evaluated at longer context lengths (400, 500, and 600 timesteps) without retraining, the baseline ensemble exhibited clear degradation while RoPE-BERT remained stable or improved at 400 and degraded less at 500 and 600. The relative advantage of RoPE over the baseline grew with extrapolation, consistent with the hypothesis that encoding relative rather than absolute position supports temporal generalization.

These findings indicate that the proposed architecture and positional encoding strategy are well suited for continuous, longitudinal BFRB monitoring where gesture duration and context length vary. The framework provides a step toward objective, privacy-preserving assessment of BFRB-

related behaviors from wearable sensor data, with the potential to complement self-report and clinician-rated measures in research and clinical practice.

3.9 Limitations

Several limitations should be considered when interpreting the results. First, all experiments were conducted on a single dataset with a fixed set of gesture classes, demographics, and sensing hardware. Generalization to other populations (e.g., different age groups or clinical cohorts), other BFRB subtypes, or different wrist-worn platforms has not been established and would require additional data and evaluation.

Second, the context lengths used for extrapolation (400, 500, 600 timesteps) span a limited range beyond the training length (300). Performance at much longer or shorter context lengths, or under different preprocessing or segmentation choices, may differ. The dataset also exhibits variable sequence lengths and class imbalance; although subject-wise splitting reduces identity leakage, the distribution of gesture types and sequence lengths may not reflect all real-world deployment scenarios.

Third, the study does not include clinical validation. Accuracy and F1 are reported at the gesture level; the relationship between these metrics and clinically meaningful outcomes (e.g., symptom severity, treatment response) has not been assessed. Deployment for clinical or therapeutic use would require further validation against clinician or self-report measures and consideration of acceptable sensitivity and specificity for the intended application.

Fourth, the framework assumes standardized data collection (e.g., sensor placement, sampling rate, and preprocessing) as in the dataset. Performance in the presence of sensor failure, significant placement variation, or different environmental conditions is unknown. Finally, the computational cost of the transformer encoder and ensemble may constrain deployment on highly resource-limited or always-on wearable devices; optimization for edge or low-power settings was not addressed here.

CHAPTER 4

Conclusion and Future Work

This report developed temporal intelligence for wearable health through gesture-based AI frameworks, addressing respiratory monitoring, asthma management, and BFRB recognition. This chapter summarizes the main contributions, outlines limitations, and discusses future directions.

4.1 Summary of Contributions

Three main lines of work were presented. First, *Coughalytics* introduced an AC gesture spotting system on commodity smartwatches. By combining IMU streams with wLCSS template matching and few-shot learning, the framework detects cough-related hand gestures before cough completion, activating the microphone only during the event to preserve privacy and storage. Evaluated in-the-wild on four participants across multiple environments, the system achieved an average accuracy of 87% with high sensitivity and specificity, demonstrating that lightweight, personalized gesture spotting is feasible on wrist-worn devices for longitudinal cough monitoring.

Second, *InhaleSense* extended gesture-based sensing to asthma management by jointly detecting cough and inhaler-use gestures from smartwatch accelerometer and gyroscope data. A compact EfficientNet-based architecture with band-pass filtering and sliding-window segmentation was

trained and refined using data augmentation, guard intervals, and overlapping windows. In evaluations with five participants, the framework achieved approximately 98% accuracy for cough detection and 99% for inhaler detection, supporting dual monitoring of symptom frequency and medication adherence in a privacy-preserving, wearable form factor.

Third, a *hybrid TCN–Transformer* framework with RoPE was proposed for BFRB-related hand gesture recognition from multimodal wrist-worn data (IMU, thermal, and time-of-flight sensors). TCNs with squeeze-and-excitation attention extract hierarchical features; a BERT-style encoder with RoPE captures long-range temporal context and generalizes to variable-length sequences. In subject-wise 5-fold cross-validation on the Child Mind Institute dataset (574,945 samples, 81 participants, 18 gesture classes), the RoPE-BERT ensemble outperformed the sinusoidal baseline by about 2.7–2.9 percentage points in accuracy and F1 at the training context length (300 timesteps) and exhibited substantially better extrapolation to longer contexts (400–600 timesteps) without retraining. These results indicate that relative positional encoding and hybrid temporal modeling are well suited for free-living, longitudinal behavioral monitoring.

Across all three contributions, the emphasis was on privacy-preserving sensing (no continuous audio/video), practical deployment on wrist-worn devices, and robustness to inter- and intra-subject variability and variable temporal structure.

4.2 Limitations and Open Questions

Several limitations remain. Dataset scale and diversity are limited in the cough and asthma studies; larger and more demographically and clinically varied cohorts would strengthen generalizability. The BFRB framework was evaluated on a single dataset and hardware configuration; transfer to other sensors, populations, and BFRB subtypes is unvalidated. Clinical validation is absent: reported metrics are at the gesture or event level and have not been linked to clinician-rated severity, treatment response, or patient-reported outcomes. Computational and energy costs of transformer-based and

ensemble models on resource-constrained wearables were not fully characterized, and on-device training for personalization was not implemented. Finally, the scope of null gestures and real-world confounders in free-living settings is not exhaustive, and false positive/negative trade-offs for specific clinical or user-facing applications remain to be defined.

4.3 Future Directions

4.3.1 Scaling, Personalization, and Clinical Validation

Future work should target larger and more diverse datasets, including clinical cohorts (e.g., asthma, COPD, BFRB patients) and longer longitudinal recordings. On-device or server-assisted personalization (e.g., few-shot adaptation of templates or classifier heads) could improve robustness across individuals and environments. Rigorous clinical validation is needed: alignment of automated gesture or event counts with clinician assessments and patient diaries, and studies of sensitivity/specificity and acceptability for use in symptom tracking, exacerbation prediction, and intervention feedback.

4.3.2 LLMs for Summarized Health and Activity Reporting

A promising direction is to combine wearable-derived gesture and event streams with JEPA or LLMs to generate *summarized, detailed reports* of health and daily activity. JEPA-style models learn representations by predicting latent states of one modality from another (e.g., from sensor embeddings to contextual or semantic embeddings) without reconstructing raw signals, which can yield compact, task-relevant representations of long time series. Such representations could be fed into decoders or lightweight language models to produce natural-language summaries (e.g., "Increased cough frequency in the evening; two inhaler uses; multiple BFRB-like gestures in the afternoon") or structured reports for clinicians and users. LLMs, when conditioned on aggregated

statistics, event logs, or learned embeddings from the wearable pipeline, could further support question-answering, trend explanation, and personalized recommendations. Careful design would be required to preserve privacy (e.g., no raw audio, minimal identifiable data), control hallucination, and validate clinical utility. Together, JEPA and LLMs could bridge low-level sensor analytics and high-level, interpretable health and activity summaries, extending the impact of the gesture-based frameworks presented in this dissertation beyond detection to communication and decision support.

4.3.3 Extension to Other Behaviors and Modalities

The same temporal modeling principles causal feature extraction, attention over variable-length context, and relative positional encoding could be applied to other health-relevant behaviors (e.g., tremor, falls, sleep-related movements) and to additional modalities (e.g., continuous glucose monitoring) where longitudinal, privacy-preserving monitoring is valuable. Integration with existing digital health platforms and electronic health records could support closed-loop feedback and population-level research while maintaining user control over data sharing and summary granularity.

In summary, this dissertation demonstrated that temporal intelligence on wrist-worn devices can support robust, privacy-preserving gesture-based monitoring for cough, asthma-related actions, and BFRB-like behaviors. Future work that scales these systems, validates them clinically, and connects them to JEPA- or LLM-based summarization and reporting will be essential to turn these capabilities into actionable tools for patients and clinicians.

Bibliography

[1] J. Jones. (2023) Networks (2nd ed.). Online. [Online]. Available: https://www.who.int/teams/noncommunicable-diseases/ncds-management/chronic-respiratory-diseases-programme#:~:text=Approximately%20half%20a%20billion%20people,%2Dincome%20countries%20(LMICs).

[2] R. T. Bhowmik and S. P. Most, "A personalized respiratory disease exacerbation prediction technique based on a novel spatio-temporal machine learning architecture and local environmental sensor networks," *Electronics*, vol. 11, no. 16, p. 2562, 2022.

[3] M. G. Crooks, A. C. den Brinker, S. Thackray-Nocera, R. van Dinther, C. E. Wright, and A. H. Morice, "Domiciliary cough monitoring for the prediction of copd exacerbations," *Lung*, vol. 199, no. 2, pp. 131–137, 2021.

[4] L. M. Nelsen, M. Kimel, L. T. Murray, H. Ortega, S. M. Cockle, S. W. Yancey, G. Brusselle, F. C. Albers, and P. W. Jones, "Qualitative evaluation of the st george's respiratory questionnaire in patients with severe asthma," *Respiratory medicine*, vol. 126, pp. 32–38, 2017.

[5] J. Andreu-Perez, H. Pérez-Espinosa, E. Timonet, M. Kiani, M. I. Girón-Pérez, A. B. Benitez-Trinidad, D. Jarchi, A. Rosales-Pérez, N. Gatzoulis, O. F. Reyes-Galaviz *et al.*, "A generic deep learning based cough analysis system from clinically validated samples for point-of-

need covid-19 test and severity levels," *IEEE Transactions on Services Computing*, vol. 15, no. 3, pp. 1220–1232, 2021.

[6] A. N. Belkacem, S. Ouhbi, A. Lakas, E. Benkhelifa, and C. Chen, "End-to-end ai-based point-of-care diagnosis system for classifying respiratory illnesses and early detection of covid-19: a theoretical framework," *Frontiers in Medicine*, vol. 8, p. 585578, 2021.

[7] M. Soliński, M. Łepek, and Ł. Kołtowski, "Automatic cough detection based on airflow signals for portable spirometry system," *Informatics in medicine unlocked*, vol. 18, p. 100313, 2020.

[8] E. L. Chuma and Y. Iano, "A movement detection system using continuous-wave doppler radar sensor and convolutional neural network to detect cough and other gestures," *IEEE Sensors Journal*, vol. 21, no. 3, pp. 2921–2928, 2020.

[9] D. Liaqat, S. Liaqat, J. L. Chen, T. Sedaghat, M. Gabel, F. Rudzicz, and E. de Lara, "Coughwatch: Real-world cough detection using smartwatches," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 8333–8337.

[10] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "uwave: Accelerometer-based personalized gesture recognition and its applications," *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 657–675, 2009.

[11] T. H. Vu, A. Misra, Q. Roy, K. C. T. Wei, and Y. Lee, "Smartwatch-based early gesture detection 8 trajectory tracking for interactive gesture-driven applications," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 1, pp. 1–27, 2018.

[12] M. H. Ko, G. West, S. Venkatesh, and M. Kumar, "Online context recognition in multisensor systems using dynamic time warping," in *2005 international conference on intelligent sensors, sensor networks and information processing*. IEEE, 2005, pp. 283–288.

[13] M. M. Jung, M. Poel, R. Poppe, and D. K. Heylen, "Automatic recognition of touch gestures in the corpus of social touch," *Journal on multimodal user interfaces*, vol. 11, no. 1, pp. 81–96, 2017.

[14] G. Laput, "Fading into the background: Unleashing ubiquitous and unobtrusive context sensing," in *Adjunct Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, 2017, pp. 87–90.

[15] A. L. Beukenhorst, K. Howells, L. Cook, J. McBeth, T. W. O'Neill, M. J. Parkes, C. Sanders, J. C. Sergeant, K. S. Weihrich, and W. G. Dixon, "Engagement and participant experiences with consumer smartwatches for health research: longitudinal, observational feasibility study," *JMIR mHealth and uHealth*, vol. 8, no. 1, p. e14368, 2020.

[16] G. Laput and C. Harrison, "Sensing fine-grained hand activity with smartwatches," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–13.

[17] P. Leamy, T. Burke, and D. Dorran, "Analysis of cough sound observed by different cough etiquettes," in *2021 32nd Irish Signals and Systems Conference (ISSC)*. IEEE, 2021, pp. 1–6.

[18] T. D. Berry and A. K. Fournier, "Examining university students' sneezing and coughing etiquette," *American journal of infection control*, vol. 42, no. 12, pp. 1317–1318, 2014.

[19] M. J.-D. Otis and J. Vandewynckel, "A many-objective simultaneous feature selection and discretization for lcs-based gesture recognition," *Applied Sciences*, vol. 11, no. 21, p. 9787, 2021.

[20] J. Smith, E. Owen, J. Earis, and A. Woodcock, "Objective measurement of cough frequency in chronic cough," *Thorax*, vol. 61, no. 5, pp. 425–430, 2006.

[21] S. S. Birring, S. Matos, R. B. Patel, B. Prudon, and D. H. Evans, "The leicester cough monitor: Preliminary validation of an automated cough detection system," *European Respiratory Journal*, vol. 31, no. 5, pp. 1013–1018, 2008.

[22] World Health Organization, "Chronic respiratory diseases programme," https://www.who.int/teams/noncommunicable-diseases/ncds-management/chronic-respiratory-diseases-programme, 2023, accessed: Oct. 6, 2023.

[23] J. Peng, C. Chen, M. Zhou, X. Xie, Y. Zhou, and C.-H. Luo, "A machine-learning approach to forecast aggravation risk in patients with acute exacerbation of chronic obstructive pulmonary disease with clinical indicators," *Scientific reports*, vol. 10, no. 1, p. 3118, 2020.

[24] S. Zeng, M. Arjomandi, Y. Tong, Z. C. Liao, and G. Luo, "Developing a machine learning model to predict severe chronic obstructive pulmonary disease exacerbations: retrospective cohort study," *Journal of Medical Internet Research*, vol. 24, no. 1, p. e28953, 2022.

[25] C.-T. Kor, Y.-R. Li, P.-R. Lin, S.-H. Lin, B.-Y. Wang, and C.-H. Lin, "Explainable machine learning model for predicting first-time acute exacerbation in patients with chronic obstructive pulmonary disease," *Journal of personalized medicine*, vol. 12, no. 2, p. 228, 2022.

[26] J. Finkelstein and I. C. Jeong, "Machine learning approaches to personalize early prediction of asthma exacerbations," *Annals of the New York Academy of Sciences*, vol. 1387, no. 1, pp. 153–165, 2017.

[27] M. G. Crooks, A. Den Brinker, Y. Hayman, J. D. Williamson, A. Innes, C. E. Wright, P. Hill, and A. H. Morice, "Continuous cough monitoring using ambient sound recording during convalescence from a copd exacerbation," *Lung*, vol. 195, no. 3, pp. 289–294, 2017.

[28] M. G. Crooks, A. C. den Brinker, S. Thackray-Nocera, R. van Dinther, C. E. Wright, and A. H. Morice, "Domiciliary cough monitoring for the prediction of copd exacerbations," *Lung*, vol. 199, no. 2, pp. 131–137, 2021.

[29] S. Birring, B. Prudon, A. Carr, S. Singh, M. Morgan, and I. Pavord, "Development of a symptom specific health status measure for patients with chronic cough: Leicester cough questionnaire (lcq)," *Thorax*, vol. 58, no. 4, pp. 339–343, 2003.

[30] N. Yousaf, K. K. Lee, B. Jayaraman, I. D. Pavord, and S. S. Birring, "The assessment of quality of life in acute cough with the leicester cough questionnaire (lcq-acute)," *Cough*, vol. 7, no. 1, p. 4, 2011.

[31] C. F. Everett, J. A. Kastelik, R. H. Thompson, and A. H. Morice, "Chronic persistent cough in the community: a questionnaire survey," *Cough*, vol. 3, no. 1, p. 5, 2007.

[32] K. M. Beeh, O. Kornmann, J. Beier, M. Ksoll, and R. Buhl, "Clinical application of a simple questionnaire for the differentiation of asthma and chronic obstructive pulmonary disease," *Respiratory medicine*, vol. 98, no. 7, pp. 591–597, 2004.

[33] J. Andreu-Perez, H. Pérez-Espinosa, E. Timonet, M. Kiani, M. I. Girón-Pérez, A. B. Benitez-Trinidad, D. Jarchi, A. Rosales-Pérez, N. Gatzoulis, O. F. Reyes-Galaviz *et al.*, "A generic deep learning based cough analysis system from clinically validated samples for point-of-need covid-19 test and severity levels," *IEEE Transactions on Services Computing*, vol. 15, no. 3, pp. 1220–1232, 2021.

[34] M. Pahar, M. Klopper, R. Warren, and T. Niesler, "Covid-19 cough classification using machine learning and global smartphone recordings," *Computers in Biology and Medicine*, vol. 135, p. 104572, 2021.

[35] W. Wei, J. Wang, J. Ma, N. Cheng, and J. Xiao, "A real-time robot-based auxiliary system for risk evaluation of covid-19 infection. arxiv 2020," *arXiv preprint arXiv:2008.07695*.

[36] C. Brown, J. Chauhan, A. Grammenos, J. Han, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring automatic diagnosis of covid-19 from crowdsourced respiratory sound data," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 3474–3484.

[37] K. McGuinness, K. Holt, R. Dockry, and J. Smith, "P159 validation of the vitalojak™ 24 hour ambulatory cough monitor," *Thorax*, vol. 67, no. Suppl 2, pp. A131–A131, 2012.

[38] M. Coyle, D. Keenan, L. Henderson, M. Watkins, B. Haumann, D. Mayleben, and M. Wilson, "Evaluation of an ambulatory system for the quantification of cough frequency in patients with chronic obstructive pulmonary disease: Cough 2005; 1: 3; published online as," *Respiratory Medicine: COPD Update*, vol. 1, no. 4, pp. 137–138, 2006.

[39] Sigma, "Opensigma: A holistic healthcare solution for ai," https://hisigma.mit.edu/home, 2022, accessed: Oct. 24, 2022.

[40] E. M. Jansen, S. J. van de Hei, B. J. Dierick, H. A. Kerstjens, J. W. Kocks, and J. F. van Boven, "Global burden of medication non-adherence in chronic obstructive pulmonary disease (copd) and asthma: a narrative review of the clinical and economic case for smart inhalers," *Journal of thoracic disease*, vol. 13, no. 6, p. 3846, 2021.

[41] J. M. Foster, T. Usherwood, L. Smith, S. M. Sawyer, W. Xuan, C. S. Rand, and H. K. Reddel, "Inhaler reminders improve adherence with controller treatment in primary care patients with asthma," *Journal of Allergy and Clinical Immunology*, vol. 134, no. 6, pp. 1260–1268, 2014.

[42] S. J. Szefler, "Monitoring and adherence in asthma management," *The Lancet Respiratory Medicine*, vol. 3, no. 3, pp. 175–176, 2015.

[43] I. Sulaiman, G. Greene, E. MacHale, J. Seheult, M. Mokoka, S. D'Arcy, T. Taylor, D. M. Murphy, E. Hunt, S. J. Lane *et al.*, "A randomised clinical trial of feedback on inhaler adherence and technique in patients with severe uncontrolled asthma," *European Respiratory Journal*, vol. 51, no. 1, 2018.

[44] M. C. Mokoka, M. J. McDonnell, E. MacHale, B. Cushen, F. Boland, S. Cormican, C. Doherty, F. Doyle, R. W. Costello, and G. Greene, "Inadequate assessment of adherence to maintenance medication leads to loss of power and increased costs in trials of severe asthma

therapy: results from a systematic literature review and modelling study," *European Respiratory Journal*, vol. 53, no. 5, 2019.

[45] P. Kadambi, A. Mohanty, H. Ren, J. Smith, K. McGuinnes, K. Holt, A. Furtwaengler, R. Slepetys, Z. Yang, J.-s. Seo *et al.*, "Towards a wearable cough detector based on neural networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2161–2165.

[46] E. C. Larson, T. Lee, S. Liu, M. Rosenfeld, and S. N. Patel, "Accurate and privacy preserving cough sensing using a low-cost microphone," in *Proceedings of the 13th international conference on Ubiquitous computing*, 2011, pp. 375–384.

[47] A. Hassan, I. Shahin, and M. B. Alsabek, "Covid-19 detection system using recurrent neural networks," in *2020 International conference on communications, computing, cybersecurity, and informatics (CCCI)*. IEEE, 2020, pp. 1–5.

[48] P. Munyard, C. Busst, R. Logan-Sinclair, and A. Bush, "A new device for ambulatory cough recording," *Pediatric pulmonology*, vol. 18, no. 3, pp. 178–186, 1994.

[49] J. Amoh and K. Odame, "Deepcough: A deep convolutional neural network in a wearable cough detection system," in *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2015, pp. 1–4.

[50] T. Drugman, J. Urbain, N. Bauwens, R. Chessini, A.-S. Aubriot, P. Lebecque, and T. Dutoit, "Audio and contact microphones for cough detection," *arXiv preprint arXiv:2005.05313*, 2020.

[51] S. J. Barry, A. D. Dane, A. H. Morice, and A. D. Walmsley, "The automatic recognition and counting of cough," *Cough*, vol. 2, no. 1, p. 8, 2006.

[52] J. I. Hall, M. Lozano, L. Estrada-Petrocelli, S. Birring, and R. Turner, "The present and future of cough counting tools," *Journal of Thoracic Disease*, vol. 12, no. 9, p. 5207, 2020.

[53] A. Ijaz, M. Nabeel, U. Masood, T. Mahmood, M. S. Hashmi, I. Posokhova, A. Rizwan, and A. Imran, "Towards using cough for respiratory disease diagnosis by leveraging artificial intelligence: A survey," *Informatics in Medicine Unlocked*, vol. 29, p. 100832, 2022.

[54] H. Al-Jahdali, A. Ahmed, A. Al-Harbi, M. Khan, S. Baharoon, S. Bin Salih, R. Halwani, and S. Al-Muhsen, "Improper inhaler technique is associated with poor asthma control and frequent emergency department visits," *Allergy, Asthma & Clinical Immunology*, vol. 9, no. 1, p. 8, 2013.

[55] S. Çakmaklı *et al.*, "An evaluation of the use of inhalers in asthma and chronic obstructive pulmonary disease," *Journal of Taibah University Medical Sciences*, vol. 18, no. 4, pp. 860–867, 2023.

[56] A. Chen, J. Zhang, L. Zhao, R. D. Rhoades, D. Y. Kim, N. Wu, J. Liang, and J. Chae, "Machine-learning enabled wireless wearable sensors to study individuality of respiratory behaviors," *Biosensors and Bioelectronics*, vol. 173, p. 112799, Feb. 2021.

[57] B. Sang, H. Wen, G. Junek, W. Neveu, L. Di Francesco, and F. Ayazi, "An accelerometer-based wearable patch for robust respiratory rate and wheeze detection using deep learning," *Biosensors*, vol. 14, no. 3, p. 118, Feb. 2024.

[58] J. Prinable *et al.*, "Derivation of respiratory metrics in health and asthma," *Sensors*, vol. 20, no. 24, p. 7134, Dec. 2020.

[59] S. Nousias, A. S. Lalos, G. Arvanitis, K. Moustakas, T. Tsirelis, D. Kikidis, K. Votis, and D. Tzovaras, "An mhealth system for monitoring medication adherence in obstructive respiratory diseases using content-based audio classification," *IEEE Access*, vol. 6, pp. 11 871–11 882, Feb. 2018.

[60] G. Greene and R. W. Costello, "Personalizing medicine—could the smart inhaler revolutionize treatment for copd and asthma patients?" *Expert Opinion on Drug Delivery*, vol. 16, no. 7, pp. 675–677, Jul. 2019.

[61] U.S. Department of Veterans Affairs, "Chronic obstructive pulmonary disease (copd): A va clinician's guide," https://www.pbm.va.gov/AcademicDetailingService/Documents/Academic_Detailing_Educational_Material_Catalog/COPD_ProviderGuide_IB101155.pdf, 2019, accessed: Month Day, Year.

[62] International OCD Foundation, "Body-focused repetitive behaviors (bfrbs)," https://iocdf.org/about-ocd/related-disorders/body-focused-repetitive-behaviors/, 2025, accessed: Dec. 22, 2025.

[63] K. E. Barber, I. F. Cram, E. C. Smith, L. K. Capel, I. Snorrason, and D. W. Woods, "Anxiety and body-focused repetitive behaviors: A systematic review and meta-analysis of comorbidity rates and symptom associations," *Journal of Psychiatric Research*, vol. 181, pp. 80–90, 2025.

[64] S. Moritz, S. Schmotz, L. Hoyer, and A. Abramovitch, "Motives for performing body-focused repetitive behaviors (bfrbs): similarities to and differences from non-suicidal self-injurious and stereotypic movement behaviors," *Cognitive Therapy and Research*, vol. 48, no. 6, pp. 1248–1254, 2024.

[65] K. Solley and C. Turner, "Prevalence and correlates of clinically significant body-focused repetitive behaviors in a non-clinical sample," *Comprehensive psychiatry*, vol. 86, pp. 9–18, 2018.

[66] Mayo Clinic Staff, "What are body-focused repetitive behaviors and how are they treated?" https://communityhealth.mayoclinic.org/featured-stories/body-focused-repetitive-behaviors/, 2021, accessed: 2025-12-22.

[67] H. G. Okumuş and D. Akdemir, "Body focused repetitive behavior disorders: behavioral models and neurobiological mechanisms," *Turkish Journal of Psychiatry*, vol. 34, no. 1, p. 50, 2022.

[68] Trichotillomania Learning Center, "Can body-focused repetitive behaviors lead to other health issues?" https://www.bfrb.org/post/can-bfrbs-lead-to-other-health-issues, 2024, accessed: 2025-12-22.

[69] S. Moritz, C. Gallinat, S. Weidinger, A. Bruhns, D. Lion, I. Snorrason, N. Keuthen, S. Schmotz, and D. Penney, "The generic bfrb scale-8 (gbs-8): a transdiagnostic scale to measure the severity of body-focused repetitive behaviours," *Behavioural and Cognitive Psychotherapy*, vol. 50, no. 6, pp. 620–628, 2022.

[70] A. Baumeister, S. Schmotz, S. Weidinger, and S. Moritz, "Is self-help dangerous? examination of adverse effects of a psychological internet-based self-help intervention for body-focused repetitive behavior (free from bfrb)," *Behavior Therapy*, vol. 55, no. 1, pp. 136–149, 2024.

[71] S. Moritz, L. N. Hoyer, N. Sarna, A. Abramovitch, C. Curran, A. S. De Nadai, and S. Schmotz, "Quo vadis dsm-6? an expert survey on the classification, diagnosis, and differential diagnosis of body-focused repetitive behaviors," *Comprehensive Psychiatry*, vol. 136, p. 152534, 2025.

[72] D. K. Ermis and E. P. Aydın, "Clinical characteristics of adolescents with body-focused repetitive behavior disorder and a validity-reliability study of the generic body-focused repetitive behavior scale-8 (gbs-8)," *Current Psychology*, vol. 44, no. 17, pp. 14 422–14 434, 2025.

[73] A. S. Mathew, A. M. Harvey, and H.-J. Lee, "Development of the social concerns in individuals with body-focused repetitive behaviors (scib) scale," *Journal of Psychiatric Research*, vol. 135, pp. 218–229, 2021.

[74] E. L. Greenberg and D. A. Geller, "Cautious optimism for a new treatment option for body-focused repetitive behavior disorders," pp. 325–327, 2023.

[75] S. Moritz, S. Weidinger, and S. Schmotz, "'free from bfrb': efficacy of self-help interventions for body-focused repetitive behaviors conveyed via manual or video," *Journal of Contemporary Psychotherapy*, vol. 54, no. 2, pp. 103–112, 2024.

[76] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[77] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019, pp. 4171–4186.

[78] K. Alomar, "Cnns, rnns and transformers in human activity recognition," *Artificial Intelligence Review*, 2025.

[79] Y. Jin, L. Hou, and Y. Chen, "A time series transformer for fault diagnosis," *arXiv preprint arXiv:2108.12562*, 2021.

[80] J. E. Grant, S. Valle, I. H. Aslan, and S. R. Chamberlain, "Clinical presentation of body-focused repetitive behaviors in minority ethnic groups," *Comprehensive Psychiatry*, vol. 111, p. 152272, 2021.

[81] S. Moritz, D. Penney, K. Ahmed, and S. Schmotz, "A head-to-head comparison of three self-help techniques to reduce body-focused repetitive behaviors," *Behavior Modification*, vol. 46, no. 4, pp. 894–912, 2022.

[82] S. Moritz, L. Borgmann, A. Alfawal, and S. Schmotz, "Beyond the surface: Neurocognitive deficits in body-focused repetitive behaviors," *Journal of the International Neuropsychological Society*, pp. 1–9, 2025.

[83] V. La Buissonnière-Ariza, J. Alvaro, M. Cavitt, B. M. Rudy, S. L. Cepeda, S. C. Schneider, E. McIngvale, W. K. Goodman, and E. A. Storch, "Body-focused repetitive behaviors in youth with mental health conditions: A preliminary study on their prevalence and clinical correlates," *International Journal of Mental Health*, vol. 50, no. 1, pp. 33–52, 2021.

[84] S. B. Clark, S. Mouton-Odum, C. A. Flessner, E. J. Ricketts, T. S. Peris, D. D. Dougherty, D. W. Woods, D. J. Stein, C. Lochner, J. E. Grant *et al.*, "Self-report measures of sensory phenomena in body-focused repetitive behaviors: A comparison to healthy controls," *Cognitive Therapy and Research*, vol. 48, no. 1, pp. 147–155, 2024.

[85] H. G. Okumuş, M. E. Öksüzoğlu, Y. S. Çelik, S. K. Yıldırım, and M. Kaşak, "An examination of impulsivity and metacognitions in adolescents with body-focused repetitive behavior disorders," *Child Psychiatry & Human Development*, pp. 1–14, 2025.

[86] I. Snorrason, E. L. Belleau, and D. W. Woods, "How related are hair pulling disorder (trichotillomania) and skin picking disorder? a review of evidence for comorbidity, similarities and shared etiology," *Clinical psychology review*, vol. 32, no. 7, pp. 618–629, 2012.

[87] J. E. Grant and S. R. Chamberlain, "The role of compulsivity in body-focused repetitive behaviors," *Journal of psychiatric research*, vol. 151, pp. 365–367, 2022.

[88] J. E. Grant, I. H. Aslan, and S. R. Chamberlain, "Statistical predictors of psychosocial impairment in body-focused repetitive behaviors," *CNS spectrums*, vol. 27, no. 5, pp. 621–625, 2022.

[89] B. L. Searle, D. Spathis, M. Constantinides, D. Quercia, and C. Mascolo, "Anticipatory detection of compulsive body-focused repetitive behaviors with wearables," in *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction*, 2021, pp. 1–15.

[90] J. J. Son, J. C. Clucas, C. White, A. Krishnakumar, J. T. Vogelstein, M. P. Milham, and A. Klein, "Thermal sensors improve wrist-worn position tracking," *NPJ digital medicine*, vol. 2, no. 1, p. 15, 2019.

[91] S. Schmotz, E. Dilekoglu, L. Hoyer, A. Baumeister, and S. Moritz, "Self-help to reduce body-focused repetitive behaviors via video or website? a randomized controlled trial," *Cognitive Therapy and Research*, vol. 48, no. 1, pp. 94–106, 2024.

[92] M. G. Woolley, S. E. Schwartz, K. L. Morrison, and M. P. Twohig, "Dissemination trial of provider training of act-enhanced behavior therapy for trichotillomania: A waitlist controlled study," *Behavior Therapy*, 2025.

[93] J. T. Stiede, D. W. Woods, A. Idnani, and S. Kumar, "Short-term intervention complemented by wearable technology improves trichotillomania: A naturalistic single-case report," *Journal of Obsessive-Compulsive and Related Disorders*, 2022.

[94] B. L. Searle, D. Spathis, M. Constantinides, D. Quercia, and C. Mascolo, "Anticipatory detection of compulsive body-focused repetitive behaviors with wearables," in *Proceedings of the ACM MobileHCI*, 2021.

[95] L. Liu, Z. Cao, and T. Li, "Facetouch: Practical face touch detection with a multimodal wearable system for epidemiological surveillance," in *ACM/IEEE International Conference on Internet of Things Design and Implementation (IoTDI)*, 2023.

[96] C. Marini, J. C. Hermann, N. D. Gold, N. S. Northover, A. J. Mallard, R. A. Smith, and P. D. Petridis, "Manual content analysis of online customer reviews on wearable technologies for the treatment of body-focused repetitive behaviours," *Archives of Psychiatry Research: An International Journal of Psychiatry and Related Sciences*, vol. 61, no. 2, pp. 155–160, 2025.

[97] A. Alesmaeil and E. Şehirli, "Real-time nail-biting detection on a smartwatch using three cnn models pipeline," *Computational Intelligence*, vol. 41, no. 1, p. e70020, 2025.

[98] N. M. Alanazi and M. Adnan, "Quintuple validation inertial framework: Multimodal sensor fusion and deep learning for precise detection of body-focused repetitive behaviors," 2025.

[99] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 156–165.

[100] S. Bai, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[101] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan, "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting," *Advances in neural information processing systems*, vol. 32, 2019.

[102] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, "A transformer-based framework for multivariate time series representation learning," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 2114–2124.

[103] Y. Tay, M. Dehghani, D. Bahri, and D. Metzler, "Efficient transformers: A survey," *ACM Comput. Surv.*, vol. 55, no. 6, Dec. 2022. [Online]. Available: https://doi.org/10.1145/3530811

[104] O. Press, N. A. Smith, and M. Lewis, "Train short, test long: Attention with linear biases enables input length extrapolation," *arXiv preprint arXiv:2108.12409*, 2021.

[105] A. Kazemnejad *et al.*, "The impact of positional encoding on length generalization in transformers," *arXiv preprint arXiv:2305.19466*, 2023.

[106] J. Su, Y. Lu, S. Pan, B. Wen, and Y. Liu, "Roformer: Enhanced transformer with rotary position embedding," *arXiv preprint arXiv:2104.09864*, 2021.

[107] A. Chen *et al.*, "Extending context window of large language models via positional interpolation," *arXiv preprint arXiv:2306.15595*, 2023.

[108] Y. Ben Amor *et al.*, "On the impact of position bias in token classification tasks," *arXiv preprint arXiv:2304.13567*, 2023.

[109] L. Newman and M. Demkin, "Cmi - detect behavior with sensor data," https://kaggle.com/competitions/cmi-detect-behavior-with-sensor-data, 2025, kaggle.