Delft University of Technology
Master of Science Thesis in Computer and Embedded Systems Engineering

Activity Segmentation for Wireless Sensing

Marijn Sluijs

TU Delft Delft University of Technology

Activity Segmentation for Wireless Sensing

Master of Science Thesis in Computer and Embedded Systems Engineering

Embedded Systems Group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Van Mourik Broekmanweg 6, 2628 XE Delft, The Netherlands

Marijn Sluijs

24th March 2026

Author
Marijn Sluijs (m.sluijs@student.tudelft.nl)
Title
Activity Segmentation for Wireless Sensing
MSc Presentation Date
23 March 2026

Graduation Committee

Dr. Marco Antonio Zuñiga Zamalloa (chairman)	Delft University of Technology
Dr. Fransesco Fioranelli	Delft University of Technology
Dr. Arash Asadi	Delft University of Technology
Dr. Silvia Pintea	Tilburg University
Fabian Portner	Delft University of Technology

Abstract

WiFi sensing enables non-intrusive, device-free monitoring of human activities by analyzing Channel State Information (CSI) extracted from commodity WiFi signals. While most research has studied Human Activity Recognition on pre-segmented clips, the harder problem of temporal activity segmentation — partitioning a continuous CSI stream into labeled activity intervals — has received less attention, and progress is limited by the absence of high-quality datasets and standardized evaluation infrastructure.

This thesis addresses that gap through three interconnected contributions. First, we introduce WiPos, a multimodal dataset in which a subject performs activities at freely varying positions, annotated with millisecond-scale precision using motion capture. Second, we present Breaking-CSI, a unified benchmarking framework that enables fair, reproducible comparison of segmentation methods across multiple datasets. Third, we propose DopplerTAS, a temporal activity segmentation model that operates on Doppler features derived from the time-differential CSI phase rather than raw amplitude, making predictions largely position-invariant.

Experiments using Breaking-CSI to evaluate representative baselines from the literature show that all of them suffer a consistent accuracy drop on WiPos compared to their native datasets, confirming that positional variation is the dominant challenge. DopplerTAS achieves 96.7% frame accuracy and 90.4% mIoU on WiPos, improving over 30 percentage points on both metrics.

Together, these contributions provide the dataset quality, evaluation thoroughness, and modeling approach needed to advance WiFi-based temporal activity segmentation from isolated recognition experiments toward continuous, position-robust sensing.

Preface

WiFi signals fill every room, reflecting off walls and rippling around moving people. The idea that this invisible field encodes enough information to determine what someone is doing and when — without a camera or wearable — struck me as both interesting and practically important. What drew me to the segmentation problem specifically was that it remains comparatively underexplored in WiFi sensing: most work addresses recognition on pre-segmented data, leaving the harder question of where one action ends and the next begins largely open — one that would not have been possible without the CSI toolkit released by Halperin et al. [18] in 2011, which first made fine-grained CSI measurements accessible on commodity hardware.

I would like to thank my supervisors Arash Asadi, Silvia Pintea, and Fabian Portner for their guidance and constructive feedback throughout the project, and the other members of my graduation committee — Marco Zuñiga and Fransesco Fioranelli — for their time and engagement with this work.

I would also like to thank Qing Wang, who first pointed me toward the available projects in the Embedded System group. That conversation led directly to this thesis topic.

I thank Anne-Sophie Straathof and Wendy Hu for introducing me to the PhaseSpace motion capture system and for patiently explaining how it works.

Marijn Sluijs

Delft, The Netherlands
24th March 2026

Contents

Chapter 1

Introduction

1.1 Motivation

WiFi sensing harnesses ubiquitous WiFi signals to enable non-intrusive, device-free monitoring of environments and human activities. Early works in device-free sensing analyzed changes in Received Signal Strength Indicator (RSSI) to detect environmental changes and human presence [68]. However, RSSI provides only coarse-grained information, limiting the granularity of sensing applications. A significant advancement came with the exploitation of Channel State Information (CSI), which provides fine-grained physical layer measurements across multiple subcarriers and antenna pairs [18]. Seminal works [57, 42, 3] demonstrated that changes in the wireless propagation environment, caused by human motion, gestures, and even physiological signals, can be detected and analyzed through CSI variations. Unlike RSSI, CSI captures both amplitude and phase information at the subcarrier level, enabling more precise and diverse sensing capabilities.

The field of WiFi sensing has matured significantly in recent years, transitioning from academic research to practical deployment and standardization. The IEEE 802.11bf standard [13], currently under development, aims to standardize WiFi sensing functionalities by exposing CSI to authenticated devices in WiFi networks, recognizing its growing importance. Commercial products have begun to emerge, such as Origin Wireless's "Time Reversal" technology [6] for presence detection and fall detection, and WiFi-based intrusion detection systems [72]. Beyond WiFi, similar sensing techniques are being investigated; for example in 6G networks, where integrated sensing and communication (ISAC) is envisioned as a core capability [33]. This growing ecosystem underscores WiFi sensing as a cost-effective, scalable solution for smart homes, healthcare monitoring, security, and industrial applications, leveraging existing communication infrastructure without requiring specialized hardware.

Despite these advances, most WiFi sensing research has focused on Human Activity Recognition (HAR), where the goal is to classify isolated activities from sensor data [36]. These works typically assume pre-segmented input: fixed-length windows of CSI values, each containing exactly one activity, which are then classified by recognition models. This assumption simplifies the problem but ignores a critical preceding challenge: how to properly achieve such seg-

mentation in the first place. The problem of temporal activity segmentation — automatically partitioning continuous CSI streams into meaningful activity segments — has received far less attention [10, 4, 69].

This gap is particularly problematic for real-world deployment. Practical WiFi sensing applications receive continuous streams of raw CSI data, not pre-cut segments. Real-world systems must understand ongoing human behavior as it unfolds, detecting when one activity ends and another begins without manual intervention. Most existing CSI-based systems assume clean, pre-labeled windows, which is unrealistic in practical scenarios where activities occur continuously, transitions are ambiguous, and activity durations vary significantly. Moreover, CSI data presents unique segmentation challenges compared to other modalities such as video. WiFi signals are inherently noisy, affected by environmental interference, hardware imperfections, and multipath propagation. Unlike visual data where object boundaries are often perceptually clear, CSI patterns are abstract representations of physical phenomena, making temporal boundaries between activities difficult to spot even for domain experts.

Where segmentation is addressed, it has traditionally been done on thresholding, typically based on energy variations in CSI amplitude [55, 27, 61]. This approach is fragile and unreliable: it can only segment activities that produce large amplitude changes, fails when transitions are gradual or subtle, and requires careful tuning of environment-dependent thresholds. Recently, several works have begun applying machine learning techniques to CSI-based activity segmentation [62, 34, 69]. While these methods show promise and improve upon threshold-based approaches, they still face significant limitations. Performance remains suboptimal for complex scenarios where activity boundaries are not predefined, and the signal is continuous and untrimmed [34, 69]. Furthermore, the methods struggle with multi-person detection and deployment on resource-constraint devices, limiting the practical usability [69].

Current datasets and segmentation models present several bottlenecks that hinder advancement. Most publicly available CSI datasets are small-scale, with limited subjects (typically 3–5 individuals), moderate sampling rates (50-100 Hz), restricting model generalization and the capture of rapid activity transitions [16]. Beyond scale, existing datasets often suffer from problematic annotation practices: temporal boundaries are derived from manual video labeling, introducing human error and imprecision at the millisecond scale where CSI operates [11, 16]. Many datasets provide only isolated activity clips rather than continuous activity streams with natural transitions [11, 62, 34], failing to reflect realistic deployment scenarios. Activities are often performed with uniform duration and style, lacking the natural variability present in real-world behavior. Furthermore, some datasets employ intransparent or inconsistent preprocessing pipelines, making it difficult to reproduce results or fairly compare methods. These data quality issues create a fundamental ceiling on model performance; even sophisticated deep learning architectures cannot overcome inaccurate or unrealistic training data. Existing segmentation approaches struggle with fine-grained activities in continuous scenarios. While threshold-based methods with sliding windows work for coarse-grained tasks like fall detection, they cannot distinguish fine-grained activities with similar signal characteristics, such as different gestures [69]. Many approaches instead rely on pauses between activities to mark segmentation boundaries [69], an unrealistic constraint for real-world applications. Learning-based methods can overcome these limitations when

trained on large, diverse datasets with activities at multiple granularities [9], but such datasets are currently scarce in WiFi sensing [34].

This thesis addresses these challenges by creating the dataset, benchmarking infrastructure, and modeling approach needed to advance WiFi-based activity segmentation. The core contribution is a high-quality, multi-modal dataset built from the ground up: a purpose-built recording environment merges CSI measurements from multiple commercial routers with synchronized motion capture and video data, while purpose-built annotation tools leverage motion capture to achieve millisecond-accurate temporal labels, precision unattainable through manual video labeling. To enable rigorous evaluation of segmentation methods, a unified benchmarking framework was developed that integrates existing approaches into a shared environment with standardized data formats, preprocessing pipelines, and evaluation metrics. Building on this infrastructure, we propose DopplerTAS, a temporal activity segmentation model that operates on Doppler features derived from the time-differential CSI phase rather than raw amplitude, making its predictions largely invariant to the subject's position. This approach provides the foundation for bridging the gap between isolated activity recognition and continuous activity understanding, moving WiFi sensing toward reproducible research and practical deployment.

1.2 Problem Statement

Building on the motivation outlined above, this thesis addresses two interconnected gaps that hinders progress in WiFi-based activity segmentation: *the lack of high-quality, accurately annotated datasets and standardized evaluation frameworks necessary for developing and comparing segmentation methods,* and *the inability of existing models to generalize across spatial positions due to their reliance on position-dependent raw CSI amplitude.*

While algorithmic approaches to temporal activity segmentation have been proposed, their development and evaluation are constrained by three interconnected infrastructure challenges and one fundamental modeling challenge:

Data Quality Challenges Existing CSI-based activity datasets suffer from critical limitations that undermine their utility for segmentation research. Manual video-based annotation introduces temporal imprecision, creating a fundamental ceiling on achievable segmentation accuracy; models cannot learn boundaries more precisely than the ground truth labels. Datasets often consist of isolated activity clips rather than continuous streams with natural transitions, failing to capture the realistic activity boundaries from ongoing motion. Additionally, many datasets lack synchronized auxiliary modalities (e.g., motion capture, video) that could enable more precise labeling or serve as complementary inputs for multimodal approaches.

Reproducibility and Comparison Challenges The absence of standardized benchmarking infrastructure prevents fair comparison across segmentation methods. Works employ different datasets, preprocessing pipelines, train-test splits, and evaluation metrics, making it impossible to determine which algorithmic contributions genuinely improve performance versus benefiting from favorable experimental conditions. Without unified evaluation protocols, the

field cannot systematically assess progress or identify which approaches are most effective for activity segmentation.

Evaluation Methodology Challenges Segmentation evaluation requires metrics that assess both temporal localization accuracy (boundary precision) and classification correctness, yet protocols vary across works. Boundary tolerance thresholds, handling of partial overlaps, and metric choices differ, complicating interpretation of reported performance and preventing meaningful comparison across studies.

Position-Induced Generalization Challenges Existing WiFi activity segmentation models process raw CSI amplitude, which encodes the full multipath channel determined by the subject's motion and their position relative to the antennas. As a result, even the same activity performed at a different location in the room produces a different CSI pattern. Models trained on datasets where subjects remain at a fixed position therefore fail to generalize to recordings with spatial diversity, limiting practical deployment.

1.2.1 Research Questions

This thesis investigates the infrastructure and modeling approaches necessary for advancing WiFi-based activity segmentation:

1. **What dataset characteristics enable rigorous segmentation research?** Specifically, what annotation precision is required? What role do synchronized multimodal data play? How should continuous activity sequences be structured?

2. **How can standardized benchmarking enable fair model comparison?** What preprocessing, data formatting, and evaluation protocols are needed to assess segmentation methods consistently across different approaches?

3. **What evaluation methodologies appropriately assess segmentation quality?** Which metrics capture both temporal precision and classification accuracy? How should these metrics be applied to enable meaningful comparison?

4. **Can Doppler features derived from CSI phase enable position-invariant activity segmentation?** Specifically, does operating on time-differential phase rather than raw amplitude substantially improve segmentation accuracy when subjects perform activities at diverse positions in the room?

Addressing these questions requires contributions in data collection methodology, annotation infrastructure, a benchmarking framework, and model design. This thesis delivers each of these contributions, as detailed in Section 1.3.

1.3 Contributions

The main contributions of this thesis are summarized as follows:

1. **WiPos dataset:** A multimodal dataset comprising WiFi CSI, motion capture data, and video recordings, annotated with millisecond-precise activity segmentation labels. One subject performed ten activities at 16 positions throughout the recording space, providing spatial diversity absent from most existing CSI datasets.

2. **Annotation infrastructure:** A custom annotation tool that presents synchronized high-rate motion capture alongside video, enabling an annotator to place activity boundaries with millisecond-scale precision that manual video labeling alone cannot provide.

3. **Breaking-CSI benchmarking framework:** A unified benchmarking framework for WiFi CSI-based activity segmentation that standardizes data preprocessing, model interfaces, and evaluation metrics, enabling fair, reproducible, and cross-dataset comparison of segmentation models.

4. **DopplerTAS:** A novel model for WiFi-based temporal activity segmentation that operates on Doppler features derived from the time-differential channel phase, making predictions position-invariant. DopplerTAS uses a bidirectional LSTM with a per-frame classifier, achieving 90.6% frame accuracy on the WiPos dataset.

5. **Experimental evaluation:** A comprehensive evaluation of existing segmentation models and DopplerTAS, conducted using the benchmarking framework on both the WiPos dataset and existing public CSI datasets.

1.4 Thesis Outline

This thesis is structured as follows.

Chapter 2 introduces the necessary background on WiFi Channel State Information (CSI) and activity segmentation, and reviews related work in CSI-based human activity analysis, relevant datasets and benchmarks.

Chapter 3 presents the WiPos dataset. It details the experimental setup, data collection process, synchronization of modalities, and the labeling methodology used to obtain millisecond-precise activity annotations.

Chapter 4 describes the proposed Breaking-CSI benchmarking framework, including its design principles, architecture, and the methods used to enable fair and reproducible evaluation of segmentation models across multiple datasets.

Chapter 5 introduces DopplerTAS, a novel model for WiFi-based temporal activity segmentation. It motivates the use of Doppler features for position-invariant sensing, describes the BiLSTM-based architecture, and presents the training protocol.

Chapter 6 presents the experimental evaluation. It reports benchmarking results for existing segmentation models across all datasets, and evaluates DopplerTAS on the WiPos dataset including ablation studies on temporal context and spatial diversity.

Chapter 7 discusses the key findings of the experimental evaluation, analyzes the strengths and limitations of the WiPos dataset, DopplerTAS, and benchmarking framework, and reflects on their implications for CSI-based activity segmentation.

Finally, Chapter 8 summarizes the contributions of this thesis and outlines directions for future research.

Chapter 2

Background and Related Work

This chapter provides the technical foundations and related work necessary for the rest of the thesis. Section 2.1 introduces WiFi Channel State Information (CSI): its physical meaning, measurement model, and the signal processing that relates raw CSI to human motion. Section 2.2 distinguishes activity recognition from temporal activity segmentation and summarizes the challenges specific to the segmentation task. Section 2.3 surveys existing CSI-based sensing datasets, segmentation methods, and benchmarking efforts, identifying the gaps that motivate this work.

2.1 Channel State Information

2.1.1 CSI Fundamentals

Modern WiFi systems based on the IEEE 802.11 protocol have evolved to support higher data rates through two key techniques: Orthogonal Frequency-Division Multiplexing (OFDM) and Multiple-Input Multiple-Output (MIMO) [20, 38]. OFDM divides bandwidth into multiple orthogonal subcarriers for robust frequency-diverse transmission, while MIMO exploits spatial diversity through multiple antennas to increase throughput and reliability [38].

These systems fundamentally require channel estimation to enable communication. Wireless communication lacks inherent synchronization between transmitter and receiver, necessitating synchronization mechanisms built into the protocol through preambles and pilot symbols. Each packet contains known training symbols that allow the receiver to estimate the wireless channel, enabling equalization — the removal of channel distortion — to recover transmitted data [18]. This per-packet channel estimation occurs within a channel coherence interval, during which the environment is assumed sufficiently static to maintain a constant channel response. Researchers have recently recognized that these channel estimates are highly correlated with changes in the physical propagation environment [3, 36]. The same channel characteristics that must be compensated for communication contain rich information about environmental conditions, including human presence and movement. This has enabled a new

paradigm: repurposing CSI, originally extracted for communication purposes, for device-free sensing and activity inference.

Channel State Information is obtained through a channel estimation procedure using known preamble symbols transmitted at the start of each WiFi packet [18]. This estimation reveals the channel response, which describes how the wireless channel affects transmitted signals. For a system with N_t transmit antennas, N_r receive antennas, and N_s subcarriers, CSI can be represented as a complex-valued matrix $\mathbf{H} \in \mathbb{C}^{N_r \times N_t \times N_s}$, where each element $H_{i,j,k}$ represents the channel response from transmit antenna j to receive antenna i at subcarrier k. The relationship between transmitted and received symbols on subcarrier k is given by:

$$y_k = H_k \cdot x_k + \eta \tag{2.1}$$

where y_k is the received symbol, H_k is the channel response at frequency f_k, x_k is the transmitted symbol on subcarrier k and η is some noise [5]. In MIMO systems, this extends to matrix form accounting for all antenna pairs.

CSI characterizes how wireless signals propagate from transmitter to receiver, capturing the combined effects of multipath propagation, shadowing, scattering, reflection, and diffraction in the environment [20]. In indoor environments, signals typically reach the receiver via multiple propagation paths, each experiencing different reflections, amplitude changes, and delays. The received signal can be modeled as the superposition of these paths:

$$H_k = \sum_{i=1}^{N_p} a_i e^{-j2\pi f_k \tau_i} \tag{2.2}$$

where N_p is the number of propagation paths, a_i is the complex amplitude (amplitude and phase shift) of path i due to reflections, f_k is the frequency of subcarrier k, and τ_i is the propagation delay of path i [5]. Each path introduces a custom amplitude scaling and phase shift depending on the number of reflections, the materials encountered, and the path length.

Each CSI measurement H_k results from the interference of all propagation paths on the complex plane. The observed amplitude $|H_k|$ and phase $\angle H_k$ arise from constructive and destructive interference between paths — paths with similar phase reinforce, while opposing phases can cancel, potentially producing zero amplitude even without attenuation. When objects or people move in the environment, the path lengths τ_i and amplitude a_i change over time, causing variations in CSI. Notably, changes in path length introduce time-dependent phase shifts in the path components of the channel. A time-varying phase shift manifests the same formulaic structure as a frequency shift — the Doppler frequency — since both represent frequency shifts due motion. This is why human movement creates characteristic Doppler signatures in CSI measurements. These fine-grained physical layer measurements provide significantly richer frequency-resolved information than coarse-grained Received Signal Strength Indicator (RSSI) values, enabling more sophisticated sensing applications [5].

2.1.2 CSI for WiFi Sensing

WiFi sensing leverages CSI measurements to monitor and interpret changes in the wireless propagation environment. Unlike coarse-grained RSSI that provides

one value per packet, CSI is a fine-grained physical layer measurement describing amplitude and phase on each subcarrier [60]. Due to multipath effects, subcarriers travel along different fading and scattering paths, creating frequency diversity with unique amplitude and phase characteristics. This enables construction of unique signal "fingerprints" that capture environmental state [60]. When humans move, they alter propagation paths through obstruction, reflection, and scattering, with changes captured by CSI variations across subcarriers and antennas.

This sensing modality offers several practical advantages. First, WiFi sensing reuses existing communication infrastructure, eliminating the need for specialized sensing hardware and enabling cost-effective deployment in environments already equipped with WiFi access points. Second, WiFi signals penetrate walls and operate in non-line-of-sight conditions, unlike vision-based systems. Third, WiFi sensing is non-intrusive and privacy-preserving, as it does not capture visual information about individuals or their surroundings. These characteristics make WiFi sensing particularly suitable for applications in smart homes, elderly care and ambient monitoring where camera surveillance or wearable devices may be undesirable [36, 25].

CSI Sensitivity to Human Motion CSI exhibits high sensitivity to human activities due to the significant impact of the human body on wireless signal propagation. The human body acts as a substantial obstacle and reflector for WiFi signals in the 2.4 GHz and 5 GHz bands. When a person moves, multiple effects influence the received CSI: (1) dynamic multipath, where moving body parts create time-varying reflection and scattering paths, (2) Doppler shifts caused by motion velocity, and (3) shadowing effects when the body obstructs direct or indirect signal paths [57].

Different activities produce characteristic temporal patterns in CSI measurements. Figure 2.1 illustrates this through a Doppler spectrogram, which reveals the frequency shifts caused by motion: walking produces moderate, periodic Doppler patterns, standing shows minimal frequency content, and running generates strong, high-frequency Doppler shifts corresponding to rapid movement. This temporal sensitivity makes CSI suitable for inferring a wide range of information about human motion, including activity and gesture recognition, vital sign monitoring (e.g., breathing rate), gait-based identification, presence detection, and other motion-related applications.

Practical CSI Measurement Challenges

While the idealized channel model in Equation 2.2 describes the theoretical propagation environment, CSI measurements extracted from commodity WiFi hardware exhibit several practical artifacts that must be addressed for sensing applications.

Phase Offset and Synchronization Errors. The measured CSI phase contains random offsets introduced by lack of synchronization between transmitter and receiver, as well as hardware inaccuracies in the signal processing chain [73].

Most prominently, sampling time offset (STO) and carrier frequency offset (CFO) introduce a linear deviation across subcarriers, where both slope and

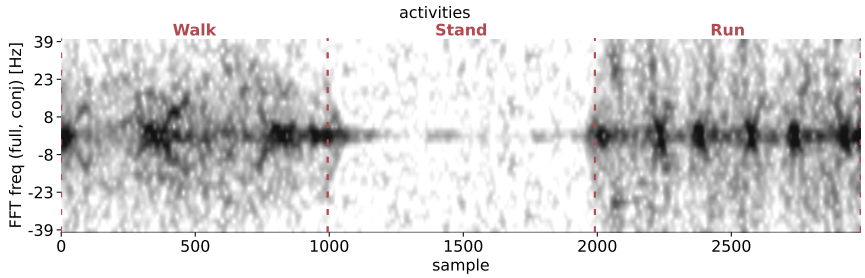


Figure 2.1: Doppler spectrogram showing characteristic motion signatures for different activities. Walking produces moderate, periodic Doppler patterns, standing exhibits minimal frequency content, and running generates strong high-frequency Doppler shifts. The vertical axis shows Doppler frequency, while pixel intensity indicates signal strength.

offset may vary randomly between packets and across antenna pairs, obscuring the true propagation phase information needed for sensing.

The measured phase can be expressed as:

$$\angle \tilde{H}_k = \angle H_k - 2\pi k f_k \delta_t + \beta + Z \tag{2.3}$$

where $\angle H_k$ is the true channel phase, δ_t is the timing offset at the receiver, f_k is the subcarrier spacing between two adjacent subcarriers, β is the total phase offset, and Z represents additive white Gauss measurement noise [73]. The linear term $2\pi k f_k \delta_t$ captures the slope across subcarriers due to STO, while β represents the constant offset.

Automatic Gain Control (AGC) Effects. WiFi receivers employ AGC to maintain signal levels within the dynamic range of the analog-to-digital converter. AGC dynamically adjusts receiver gain based on received signal strength [8], introducing multiplicative scaling factors to CSI amplitude measurements that are independent of the actual channel. Since AGC settings can change between packets, raw CSI amplitudes are not directly comparable across time, complicating activity detection that relies on amplitude variations.

Antenna Conjugation and Hardware Asymmetries. Many phase errors, particularly those introduced by STO and CFO, are shared across all receive antennas on a single device. This means that CSI measurements from all antennas at any given time share the same linear phase offset described in Equation 2.3. Antenna conjugation is a technique to remove these shared phase offsets by using one antenna as a phase reference. Through complex conjugation and multiplication, the shared linear component cancels out in the remaining antennas. When multiple receive antennas are available, this is the preferred method for phase sanitization. When only a single receive antenna is available, or when sacrificing one antenna as a phase reference is undesirable, linear phase removal can be applied instead. This technique fits and subtracts the linear component directly from the subcarrier phase measurements of a single antenna, recovering an approximation of the true channel phase without requiring a reference antenna. Hardware imperfections and device-specific CSI extraction implementations introduce additional measurement variations that may differ across WiFi chipsets.

2.2 Activity Recognition and Segmentation

This section distinguishes between two fundamental tasks in human activity analysis: activity recognition, which classifies pre-segmented activity instances, and activity segmentation, which identifies temporal boundaries between activities in continuous streams. Understanding this distinction is critical, as most existing WiFi sensing research addresses only recognition while assuming manual segmentation, limiting practical deployment.

2.2.1 Activity Recognition

Human Activity Recognition (HAR) classifies activities from sensor data [31]. In the standard formulation, each input is assumed to contain exactly one activity: a sensor reading or video clip showing a person walking, sitting, or performing another action. The classifier's task is to identify which activity the input represents.

This single-activity assumption has enabled significant algorithmic progress but introduces a fundamental dependency: continuous sensor streams must first be divided into single-activity segments. This segmentation is typically performed manually, which is time-consuming, subjective, and prevents autonomous system operation. Further, reality often does not provide pre-segmented windows of data. This is especially true for WiFi sensing, where CSI arises naturally during communication, and is not tied to an activity trigger, nor does it directly allow for activity disambiguation. Real-world deployment requires systems that can process continuous activity streams without manual intervention—a capability that recognition alone cannot provide.

2.2.2 Activity Segmentation

Activity segmentation addresses a more comprehensive problem than isolated activity recognition: automatically partitioning continuous sensor data streams into temporal segments and classifying each segment. This eliminates manual pre-segmentation and enables real-time processing of natural, continuous behavior [15].

Taxonomy of Temporal Activity Analysis Tasks Several related but distinct formulations exist for analyzing activities in continuous streams:

- **Temporal Action Segmentation (TAS):** The task of assigning an activity label to every time step in a continuous sequence, thereby solving segmentation and classification jointly. This produces dense, frame-level predictions [12]
- **Temporal Action Detection (TAD):** Outputs discrete temporal segments with start times, end times, and activity labels. This is semantically equivalent to TAS with an additional "background" or "unknown" class for non-activity periods, but focuses on sparse predictions[34].
- **Action Boundary Detection (ABD):** Focuses solely on identifying temporal boundaries between activities without classification, treating segmentation and recognition as separate tasks [14].

2.2.3 Challenges of Segmentation

Activity segmentation presents several fundamental challenges that makes it inherently more difficult than isolated activity recognition:

Temporal Boundary Ambiguity Models must identify subtle temporal boundaries between activities without explicit segmentation cues. Transitions between activities are often gradual rather than instantaneous, with boundaries that are ambiguous even to human annotators [1]. For example, the transition from "standing" to "walking" may involve preparatory movements that blur the exact boundary.

Variable Activity Duration Segmentation systems must handle activities with different temporal extents, from brief actions lasting a fraction of a second to extended activities spanning minutes [56, 69].

Activity Granularity and Hierarchical Structure What constitutes a distinct activity depends on the desired semantic granularity level, which varies by application; annotations must clearly define the intended level[69].

Inter-Subject Variability Models must be robust to variations in how different individuals perform the same activity, including differences in duration, execution speed, movement style, and body dimensions [69].

Sequential Dependencies Identifying one segment's boundary may depend on understanding surrounding context. Activities do not occur in isolation but as part of meaningful sequences, requiring models to capture temporal dependencies across extended time horizons [24].

2.3 Related Work

This section surveys work relevant to CSI-based activity segmentation. Section 2.3.1 reviews segmentation methods ranging from unsupervised motion detection to supervised temporal action segmentation, including a clustering-based algorithm adapted from the video domain. Section 2.3.2 surveys existing CSI datasets, distinguishing those suitable for segmentation evaluation from recognition-focused benchmarks. Section 2.3.4 identifies the gaps that motivate this work.

2.3.1 Activity Segmentation Methods

This subsection reviews CSI-based activity segmentation methods, organized chronologically to illustrate their evolution. The literature divides into two distinct approaches based on supervision requirements and capabilities:

Unsupervised and training-free methods operate without labeled data, typically employing threshold-based detection or clustering techniques. These approaches treat activity detection as a component within larger sensing systems. Critically, they perform motion detection — distinguishing motion from static states — but do not differentiate between activity types. This limits their

applicability to recognition tasks but makes them practical for deployment scenarios where labeled training data is unavailable.

Supervised and semi-supervised methods treat segmentation holistically, requiring labeled data to train models that simultaneously detect boundaries and recognize activity types. More advanced approaches in this category enable change detection between continuous activities without intervening static periods, addressing realistic scenarios where activities transition from one to another.

Clustering-based methods from the video domain are parameter-free algorithms originally developed for video action segmentation that require no domain-specific training and can be applied to any sequential modality, including WiFi CSI.

The transition from RSSI to CSI-based sensing enabled the evolution of the first two categories, as the fine-grained channel information of CSI provides the sensitivity necessary for accurate temporal localization and activity differentiation.

Unsupervised and Training-Free Approaches

Jiang Xiao et al. [63] proposed FIMD, a device-free indoor motion detection system using fine-grained CSI. FIMD extracts temporal and frequency diversity features from CSI measurements, then applies DBSCAN clustering to identify motion events. The approach treats motion detection as an outlier detection problem: features from static environments form dense clusters representing normal conditions, while motion causes CSI deviations that appear as outliers in the feature space. DBSCAN identifies these outliers without requiring labeled training data, enabling unsupervised motion detection. The evaluation shows that CSI-based motion detection outperforms RSS-based approaches.

MoSense [17] detects human motion by analyzing CSI amplitude fluctuations caused by movement in the environment. The system operates in two stages: first, a silence analysis model characterizes stationary periods to establish a baseline for distinguishing motion from static conditions. Second, a subcarrier selection mechanism identifies which subcarriers are most sensitive to motion by measuring similarity between subcarrier responses, filtering out those dominated by noise. This selection reduces computational complexity by processing only the most informative channels. The authors demonstrate real-time motion detection using commodity WiFi hardware without requiring specialized sensors or camera-based monitoring.

RT-Fall [55] performed real-time fall detection using commodity WiFi through automatic segmentation and classification. For segmentation, the system monitors the variance of CSI phase differences across antennas to identify finishing points of fall and fall-like activities — when variance changes exceed a threshold, the system traces back to determine the activity's starting point. For classification, RT-Fall extracts eight features from the segmented CSI data, including statistical measures, signal velocity, and two novel features: Timelag (characterizing state transition speed) and Power Decline Ratio (measuring power loss after the activity finishes). These features, extracted from both CSI amplitude and phase difference, are fed into an SVM classifier to distinguish falls from fall-like activities. This enables autonomous, real-time fall detection without manual segmentation.

WiSH [70] addresses the challenge of continuous human presence detection on resource-constrained devices. Prior WiFi sensing systems required high sampling rates (100–1000 Hz) and significant computation, making whole-day monitoring impractical. WiSH introduces a lightweight real-time detection system that operates at only 20 Hz sampling rate while maintaining $> 98\%$ accuracy with 1.5 second detection delay. The system extracts correlation features in time and frequency domains to identify human-induced CSI changes, applies a duration-based event filter to reject transient noise, and employs self-calibration using quiet periods to adapt thresholds and select stable antennas. This enables deployment on embedded devices for practical, whole-day monitoring applications.

FallDeFi [40] introduces time-frequency analysis to CSI-based fall detection, addressing limitations of prioir time-domain approaches. Systems like RT-Fall [55] relied on temporal features that struggle to distinguish falls from similar activities and are sensitive to environmental changes. FallDeFi applies STFT to generate spectrograms that reveal characteristics high-frequency, high-energy fall signatures within 2–3 second intervals. The system employs a Power Burst Curve-a technique adapted from radar fall detection-to localize high-energy events as fall candidates, then extracts spectral and temporal features from these intervals. Sequential forward selection identifies features that maintain discriminative power across multiple environments, rejecting environment-specific artifacts.

To improve motion duration estimation accuracy, MoSeFi [58] proposed a device-free and duration estimation robust human motion sensing system using WiFi signal. Based on the analysis of Mobius transform, MoSeFi constructs a novel indicator for motion detection using the shape-complementary real and imaginary parts of CSI, significantly reducing errors under short-window conditions.

Naveed Tahir et al. [52] proposed an unsupervised method for human motion detection using WiFi signals. The method uses a deep clustering model trained on appropriately-preprocessed CSI data to detect human motion in the absence of any ground labels, reducing the training overhead. By removing the need for labeled training data, this method makes a significant step towards effective WiFi sensing deployable in the real world.

Prioritizing low-latency online processing, LightSeg [9] introduced an activity granularity-aware threshold that quickly adjusts based on the granularity of the current activity. A threshold post-decision mechanism first detects the end of a segment and then decides the appropriate threshold based on the most recent activity.

Taking a fundamentally different approach, Xu Wang et al. [59] proposed a training-free and environment-independent motion segmentation system that treats CSI-derived features from an image-processing perspective. They first select subcarriers based on median variance values, then compute sliding window variance from CSI ratio data to form a two-dimensional image. To enhance motion interval homogeneity and address the overlap between dynamic and static variance values, they employ a novel quasi-envelope derived from prominent peaks. An iterative segmentation algorithm based on improved Otsu thresholding then accommodates varying motion intensities.

Supervised and Semi-Supervised Approaches

DeepSeg [62] replaces threshold-based segmentation with a learned CNN classifier. Prior methods used manually tuned amplitude thresholds to detect activity boundaries, but fine-grained activities (hand gestures) and coarse-grained activities (walking, running) exhibit vastly different amplitude ranges — a threshold suitable for one type fails for the other. DeepSeg frames segmentation as classification, training a CNN to label time bins as one of four states: static, start, motion, or end. The system incorporates a feedback mechanism where classification confidence scores guide segmentation refinement: during joint training, bins that produce high-confidence classifications receive higher weights, encouraging the segmentation model to generate boundaries that facilitate accurate classification. This eliminates manual threshold tuning and handles mixed fine- and coarse-grained activities within a unified framework.

Shuang Zhou et al. [71] proposed Wi-Monitor, a system that overcomes the common assumption in prior WiFi-based activity segmentation methods that activities are separated by static intervals. Wi-Monitor segments continuous activities by first fragmentizing CSI streams into fixed-size, non-overlapping windows. Each window is processed by a convolutional network to extract short-term activity fragmentation features (AFFs). A temporal convolutional network (TCN) processes the AFFs to compute activity continuity features (ACFs), capturing long-range dependencies needed to distinguish adjacent activities. To address over-segmentation, Wi-Monitor introduces a constraint on a learned segmentation degree, penalizing excessive predicted boundaries. These components together form a system for accurate segmentation and recognition of continuous human activities in realistic, unstructured settings.

Combining adaptive thresholding with graph neural networks, Xiaolong Yang et al. [66] introduced a window variance comparison method for segmenting multiple discontinuous human behaviors. Graph structure data is extracted from the time-frequency features of individual actions, enabling recognition through graph-based learning.

WiFiTAD [34] addresses WiFi-based activity segmentation through frequency-aware feature learning and dual pyramid fusion. The architecture employs two complementary encoder types that capture different aspects of CSI dynamics: Temporal Signal Semantic Encoders (TSSE) extract semantic information across entire activity segments, while Local Sensitive Response Encoders (LSRE) capture fine-grained temporal fluctuations for boundary localization. Each TSSE operates through two parallel branches to learn high and low-frequency signal components separately. The low-frequency branch employs a transformer with a novel Signed Mask-Attention (SMA) mechanism to extract semantic features that characterize overall activity patterns. LSREs use channel-wise windowing to compute local maxima and minima, capturing rapid fluctuations indicative of activity boundaries. The outputs of multiple TSSEs and LSREs are organized into two feature pyramids at different temporal scales. A Cross-Attention Pyramid Fusion module aligns and integrates information from both pyramids, combining semantic understanding with precise temporal localization. The fused features feed into a dual-branch prediction head that simultaneously predicts activity classes and boundary locations.

Tackling the challenge of segmenting long, continuous wireless sensing data, SEGALL [69] introduced a learning-based framework using signal-independent

deep learning coupled with active learning. SEGALL achieves accurate segmentation for fine-grained activities while reducing labeling efforts and handling interference and signal complexity inherent in wireless sensing.

Clustering-Based Approach from the Video Domain

Unsupervised temporal action segmentation is a well-studied problem in the video domain, where a rich body of literature explores clustering, self-supervised learning, and generative approaches applied to dense visual feature sequences [45, 49, 30]. Video-based methods benefit from decades of representation learning research and high-dimensional, semantically rich input features, making this a comparatively mature field. WiFi CSI data is fundamentally different: it is low-dimensional, dominated by hardware noise and environmental reflections, and lacks the spatial or texture cues that video representations rely on. Consequently, video-domain approaches are not expected to transfer directly to the CSI modality. Nevertheless, including a representative video-domain baseline is valuable to empirically confirm this inapplicability and to provide a reference point for the gap between the two domains. A comprehensive review of unsupervised video segmentation methods is beyond the scope of this work; we focus on a single representative approach.

Sarfrax et al. [45] proposed TW-FINCH (Temporally-Weighted FINCH), a parameter-free, training-free hierarchical clustering algorithm for unsupervised action segmentation. Built on the FINCH [46] framework, it represents a sequence as a directed 1-nearest-neighbor graph and modulates pairwise affinities by temporal proximity, encouraging clusters that are both semantically consistent and temporally contiguous. In this thesis, TW-FINCH is adapted to WiFi CSI by treating each CSI frame as a flattened feature vector. Since the algorithm produces unlabeled clusters, the Hungarian algorithm [29] is used to find the optimal mapping between cluster indices and ground-truth class labels, providing the best-case accuracy achievable by the discovered partition.

2.3.2 Datasets for Activity Segmentation

While numerous WiFi sensing papers introduce custom datasets, few provide publicly available, well-annotated resources suitable for activity analysis. We distinguish between datasets designed for temporal segmentation tasks — which require continuous activity streams with frame-level or segment-level boundary annotations — and those designed for isolated activity recognition. The former category is essential for evaluating segmentation models, while the latter supports classification but cannot be directly applied to segmentation without additional processing.

Segmentation-Suitable Datasets

Several publicly available datasets provide continuous activity streams with temporal annotations, making them suitable for segmentation tasks.

DeepSeg Dataset DeepSeg [62] published a CSI-based activity dataset as part of their deep learning-based segmentation framework.[1] The dataset in-

[1] `https://github.com/ChunjingXiao/DeepSeg`

cludes five subjects performing ten activities (five fine-grained and five coarse-grained) with static intervals between activity segments. Activities were captured using commodity WiFi hardware with one transmit antenna and three receive antennas at 50 Hz sampling rate across 30 subcarriers.

WiFiTAD Dataset The WiFiTAD dataset, introduced alongside the WiFiTAD model [34], contains CSI recordings of three subjects performing seven activities in one environment.[2] Activities were performed with static intervals between segments. The dataset employs a single transmit and receive antenna configuration, sampling at 100 Hz across 30 subcarriers.

MM-Fi Dataset MM-Fi [65] is a multi-modal human activity dataset that includes RGB frames, depth frames, LiDAR point clouds, mmWave radar points, and WiFi CSI data.[3] The dataset features 40 subjects performing 27 activities, including 14 daily actions and 13 clinically-suggested rehabilitation exercises. Participants performed repetitions of each activity continuously. The authors developed a customized platform based on the Robot Operating System (ROS) to capture and synchronize data from multiple sensors, enabling accurate temporal annotation. CSI was captured at 1000 Hz (downsampled to 10 Hz for multi-modal synchronization) using one transmit antenna and three receive antennas across 114 subcarriers.

Wi-Monitor Dataset As part of the Wi-Monitor system, Zhou et al. [71] created a CSI-based activity dataset featuring five subjects performing ten activities across three environments.[4] The dataset includes both fine-grained and coarse-grained activities performed continuously. The authors synchronized CSI streams with video recordings for temporal annotations, although they do not mention how the synchronization is performed. Data was captured using one transmit antenna and three receive antennas at 100 Hz sampling rate across 30 subcarriers. The released data is preprocessed: the paper describes preprocessing steps including antenna conjugation and Butterworth filtering, but these steps are absent from the released code, indicating that the distributed data is already in a processed form rather than raw CSI.

2.3.3 Recognition-Focused Datasets

Several other publicly available CSI datasets focus primarily on isolated activity recognition rather than temporal segmentation. These datasets provide pre-segmented activity instances rather than continuous streams with temporal boundaries. Notable examples used in WiFi sensing evaluations include UT-HAR [67] developed to evaluate recent advancements in human activity recognition systems, EfficientFi [64] focusing on CSI compression and recognition efficiency, and CAUTION [53], which contains walking gaits from 14 subjects for gait-based human identification. While such datasets can theoretically be used to reconstruct longer sequences by concatenating different activities, this would introduce artificial boundaries that are unnatural and thus possibly easier

[2] `https://github.com/AVC2-UESTC/WiFiTAD`
[3] `https://github.com/ybhbingo/MMFi_dataset`
[4] `https://github.com/zs-zhoushuang/Wi-Monitor`

to detect than realistic transitions. Consequently, these datasets are not suitable for the segmentation tasks considered in this thesis.

2.3.4 Research Gap

The surveyed literature reveals significant progress in CSI-based activity segmentation methods, ranging from unsupervised motion detection to supervised temporal action segmentation with continuous activity recognition. However, several critical gaps remain in the dataset infrastructure needed to rigorously evaluate these approaches.

Current CSI-based activity datasets exhibit several limitations that hinder systematic segmentation evaluation. **Imprecise annotations** resulting from manual video labeling provide only moderate temporal precision (±33-100 ms) with inherent subjectivity; hardware and software delays may further offset labeling times from true activity boundaries. **Artificial patterns** introduced by static intervals between activities and predictable activity sequences create unrealistic transition patterns that do not reflect real-world sensing scenarios. **Class imbalance** stemming from unequal activity durations and occurrence counts biases model evaluation toward overrepresented activities. **Moderate sampling rates** of 50-100 Hz may miss fine-grained temporal dynamics, particularly limiting downstream applications such as Doppler processing that require high temporal resolution. Finally, **intransparent preprocessing** in some released datasets limits usability; preprocessing steps are often not fully specified due to unreleased code, making it difficult to reproduce or adapt data for specific applications. Beyond infrastructure, existing methods do not address the sensitivity of learned CSI representations to the subject's position in the room — a fundamental challenge for generalization across recording sessions that no existing dataset is designed to expose.

These gaps motivate the contributions of this thesis, described in Chapter 3 through Chapter 5.

Chapter 3

Dataset Design and Data Collection

This chapter presents WiPos (WiFi Positional activity segmentation dataset), a multimodal dataset designed to address the annotation precision and spatial-diversity limitations of existing CSI datasets. Section 3.1 gives an overview of the dataset composition, design goals, and intended use cases. Section 3.2 describes the recording environment and hardware. Section 3.5 presents the motion-capture-based annotation methodology. Section 3.6 describes the data format. Sections 3.7 and 3.8 describe derivative datasets generated from WiPos.

3.1 Description

This section provides a detailed overview of the WiPos dataset, including its composition, intended use cases, and design rationale.

3.1.1 Motivation and Design Goals

As identified in Section 2.3.4, existing CSI datasets lack annotation precision, enforce artificial activity transitions, and suffer from class imbalance. We design WiPos to address these limitations directly. Our primary objectives are to:

- Provide millisecond-precision by using high-rate motion capture data (960 Hz) as the annotation reference, achieving temporal precision far beyond what video-based labeling can offer.
- Capture continuous activity streams without artificial static intervals, reflecting realistic deployment scenarios where activities transition naturally.
- Ensure balanced activity representation to prevent models from exploiting class imbalance and to support both segmentation and recognition tasks.
- Include diverse activity granularities — from coarse-grained full-body motions to fine-grained localized movements — to evaluate segmentation across difficulty levels.

- Capture high-rate CSI measurements (1000 Hz) with constant rate, equidistant sampling to preserve fine-grained temporal dynamics and support signal processing techniques such as STFT-based Doppler spectrograms.
- Provide multi-modal synchronized data (CSI, motion capture, video) to enable interpretability analysis and multi-modal research.
- Ensure spatial diversity through multiple receive chains, providing complementary views of activity-induced channel changes from different vantage points and enabling multi-antenna processing techniques.

These design goals translate into specific requirements for data collection: precise multi-sensor synchronization, randomized activity sequencing to prevent temporal pattern exploitation, controlled duration sampling to achieve class balance, and a recording protocol that produces continuous activity streams with natural transitions.

3.1.2 Dataset Composition

We record a multi-modal dataset of human activities performed by one subject in a controlled laboratory environment. The activity set consists of ten distinct activities spanning a range of motion characteristics and granularity levels: *stand*, *walk*, *run*, *jumping jack*, *jump*, *squat*, *raising left arm*, *raising left foot*, *stirring a pot*, and *boxing*. The subject is not constrained to a fixed position throughout the recording; activities such as walking and running produce spatial displacement across the room, and subsequent fixed-position activities (e.g., stirring a pot, raising arms) are performed at wherever the subject happens to stop, introducing realistic positional variation throughout the dataset.

To ensure balanced activity representation while maintaining natural variability, we employ a constrained randomized sampling strategy. For each activity instance, we randomly sample the duration between 3 and 10 seconds, reflecting realistic variability in activity execution. We randomize both the order and duration of activities across recording sessions to prevent models from exploiting artificial temporal patterns that would not exist in real-world deployments.

We capture three synchronized modalities for each recording session:

- **Channel State Information (CSI):** We record WiFi CSI measurements at 1000 Hz, providing fine-grained wireless signal data with millisecond-scale temporal resolution for activity segmentation.
- **Motion Capture Data:** We capture optical motion capture at 960 Hz, recording precise 3D positions of body markers throughout activity execution. This modality serves as the objective ground truth for temporal segmentation labels.
- **Video Recordings:** We record video at 30 fps, providing visual context for validation and qualitative analysis.

The motion capture modality serves as the authoritative ground-truth source for temporal labels, enabling millisecond-scale annotation precision as described in Section 3.5.

3.1.3 Intended Use Cases

We design this dataset to support the following research use cases:

- **CSI-Based Activity Segmentation:** The primary intended use case is developing and rigorously evaluating temporal activity segmentation models using WiFi CSI. The high-rate CSI measurements, combined with millisecond-precision temporal annotations derived from motion capture, enable evaluation of segmentation models under realistic conditions with continuous activity streams and natural transitions. The randomized activity order, duration, and balanced class representation make the dataset particularly suitable for assessing model robustness and generalization capabilities.
- **Human Activity Recognition** With the dataset being balanced and having annotated segments, it can easily be transformed into a classical HAR dataset, and we provide scripts to do so. As such, it is strictly more general than existing HAR datasets.
- **Localization** Prior datasets contain position mostly in the sense of a discrete grid. Through the synchronized motion capture modality, we capture more general body positioning as the subject moves throughout the room. This allows for the dataset to be used for localization, i. e. to develop methods that would determine the subject's position from CSI.
- **Interpretability and Visualization:** Researchers can use motion capture and video data to understand which body movements correspond to specific CSI patterns, aiding model interpretability and providing insights into the physical basis of CSI-based sensing.
- **Ground Truth Validation:** The motion capture modality provides an objective reference for evaluating annotation methods, comparing labeling modalities, or validating alternative ground truth sources.

By providing transparent, multi-modal data with high temporal resolution and precise annotations, this dataset aims to advance the state of the art in WiFi sensing research and enable more rigorous, reproducible evaluation of activity segmentation approaches.

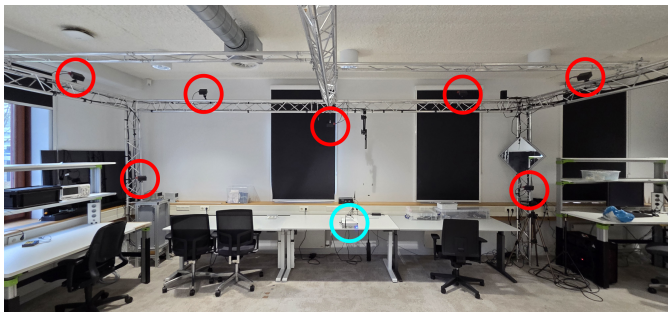
3.2 Recording Environment and Hardware Setup

This section describes the physical recording environment, hardware components, and experimental setup used for data collection. The setup is designed to capture synchronized multi-modal data with high temporal resolution and precise spatial tracking.

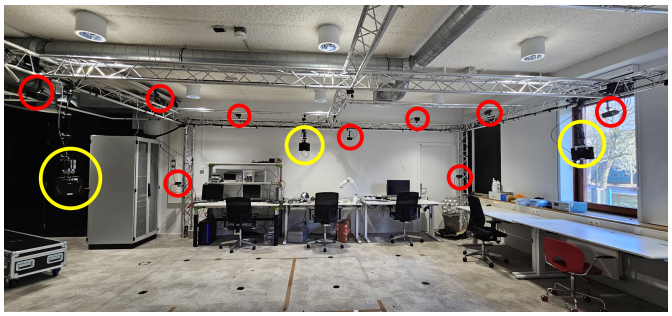
3.2.1 Recording Environment

Data collection was conducted in a controlled laboratory environment measuring approximately 6 x 6 meters. The space was configured to accommodate the WiFi sensing equipment, motion capture system, and subject movement area, shown in Figure 3.4. Figures 3.1a and 3.1b show the recording environment from two

different perspectives, illustrating the placement of all hardware components around the perimeter of the space.



(a) USRP transmitter side



(b) ASUS receivers side

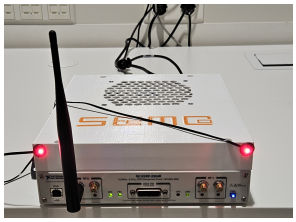
Figure 3.1: Recording environment from two perspectives. (a) View showing the USRP transmitter (highlighted in blue) positioned on a desk with PhaseSpace motion capture cameras (highlighted in red) mounted on the truss structure. (b) View showing the three ASUS receivers (highlighted in yellow) mounted on the truss structure with their antenna arrays and PhaseSpace motion capture cameras (highlighted in red).

3.2.2 WiFi CSI Capture System

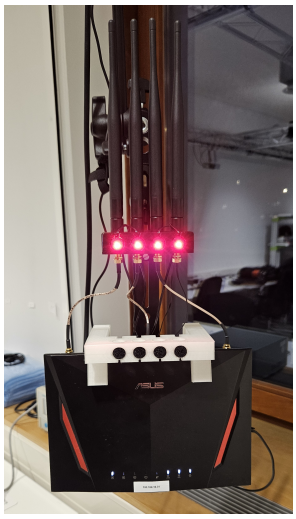
The WiFi CSI capture system comprises a software-defined radio transmitter and three commercial WiFi receivers operating in a coordinated configuration. Figure 3.2 shows the main hardware components.

Transmitter A National Instruments USRP-2954R serves as the WiFi transmitter, equipped with a single omnidirectional antenna. The USRP transmits IEEE 802.11ac VHT frames at a precise rate of 1000 Hz on channel 157 (5.785 GHz). WiFi frames are generated using the MATLAB WLAN Toolbox and transmitted on a 40 MHz channel, providing 114 usable subcarriers per antenna pair.

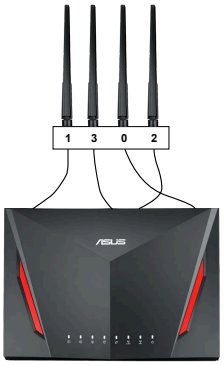
The transmitter operates with a gain of 30 dB, ensuring sufficient signal strength throughout the recording area. Two critical timing mechanisms ensure precise CSI capture: first, we implement custom synchronization code in the usrpulse framework [39] that synchronizes USRP transmission starts to within



(a) NI USRP-2954R with one antenna and two LED markers.



(b) ASUS receiver with four antennas (2 cm apart) and four LED markers.



(c) Antenna indexing for the ASUS RT-AC86U receiver.

Figure 3.2: WiFi CSI capture hardware. (a) NI USRP-2954R transmitter. (b) ASUS RT-AC86U receiver with custom antenna mount. (c) Antenna indices as they appear in the CSI data. The physical holder positions antennas left-to-right as 1, 3, 0, 2; the router port order (left-to-right) is 1, 3, 2, 0.

a millisecond of desired times. Second, the 1000 Hz transmission rate is achieved with sample-accurate timing by explicitly transmitting zero-valued IQ samples during idle periods between frames, rather than relying on operating system sleep functions. This ensures the frame rate is exact within the USRP's clock accuracy, which does not drift significantly over the short recording durations in the WiPos dataset. This deterministic, equidistant sampling spacing is critical for downstream applications that rely on uniform temporal sampling, such as Doppler-based motion analysis.

Two active LED markers from the motion capture system are mounted to the USRP housing (see Figure 3.2a), enabling precise tracking of the transmitter's spatial position throughout recordings.

Receivers Three ASUS RT-AC86U commercial WiFi routers serve as CSI receivers. Each router is equipped with four antennas, one of which is internal and was exposed through hardware modifications performed specifically for this setup. The four antennas on each router are mounted on custom 3D-printed mounts designed for this thesis, maintaining a uniform spacing of 2 cm between adjacent antennas.

The three receivers are mounted on adjustable arms attached to a truss structure. Figure 3.2b shows one receiver with its antenna array and mounted LED markers. Each receiver has four LED markers attached to track its spatial position precisely. Detailed spatial placement of the receivers relative to the

transmitter and movement area is described in Section 3.2.6.

All antennas have a physical height of 19.5 cm, providing consistent radiation patterns throughout the sensing area. The antenna indexing scheme — determining the antenna dimension ordering in all CSI data — is shown in Figure 3.2c.

Due to timestamp inaccuracies in the ASUS routers' NTP client and non-deterministic CSI extraction timing in Nexmon-CSI [48] (the open-source firmware patch used to enable CSI extraction on the ASUS routers), frame reception times are determined by matching received frame sequence numbers (incremented sequentially at the transmitter) to known USRP transmit times, as detailed in Section 3.4. This approach ensures precise alignment of CSI measurements with ground truth capture data.

Data Quality Control To ensure the data quality, the capture system monitors packet reception rate at each receiver. Recordings where any receiver captures less than 90% of transmitted frames are flagged for recapture to ensure complete CSI coverage across all devices.

3.2.3 Motion Capture System

The motion capture system employs a PhaseSpace Impulse X2E optical tracking system consisting of 16 high-speed cameras capturing at 960 frames per second. This frame rate closely matches the WiFi CSI sampling rate, enabling precise temporal alignment between modalities. The cameras are positioned around the perimeter of the recording area at various heights to provide complete coverage of the movement space with minimal occlusion. Camera placement details are presented in Section 3.2.6 and Figure 3.5.

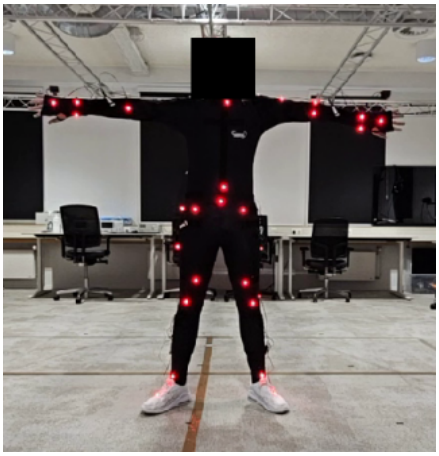
Active LED markers are placed on a full-body motion capture suit worn by the subject during recording sessions. A total of 64 markers are distributed across the head, torso, arms, and legs, providing comprehensive coverage of body movements. Figures 3.3a and 3.3b shows photographs of the marker-equipped suit from front and back views, while Figures 3.3c and 3.3d present a schematic diagram with numerical identifiers for each marker position. The complete mapping between marker identifiers (0–63) and their anatomical labels is provided in Table A.1. The high-density marker configuration enables detailed tracking of body movements and provides sufficient information to detect subtle activity transitions.

In addition to body markers, LED markers are attached to all WiFi hardware components: two markers on other USRP transmitter and four markers on each of the three receivers (see Figures 3.2a and 3.2b). Tracking the WiFi device positions allows precise knowledge of the wireless propagation geometry, enabling potential future analysis correlating body position relative to transmitter-receiver paths with observed CSI patterns.

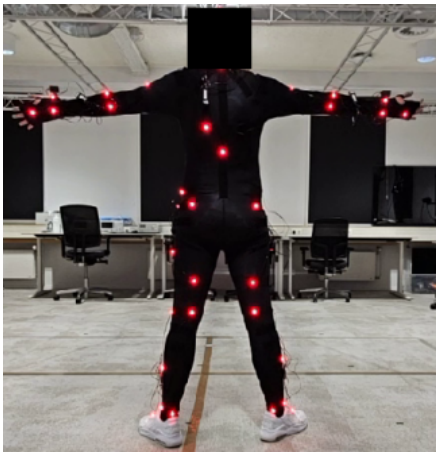
Before a recording session, the motion capture system undergoes calibration and alignment, setting the position and orientation of the cameras and the orientation of the floor, to ensure accurate 3D position tracking of all markers.

3.2.4 Video Capture System

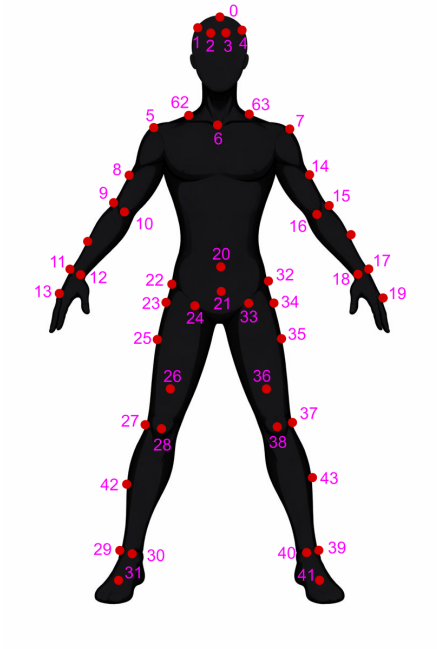
We capture synchronized video using a Samsung Galaxy S25 smartphone with the wide-angle camera lens, recording at 720p resolution and 30 frames per



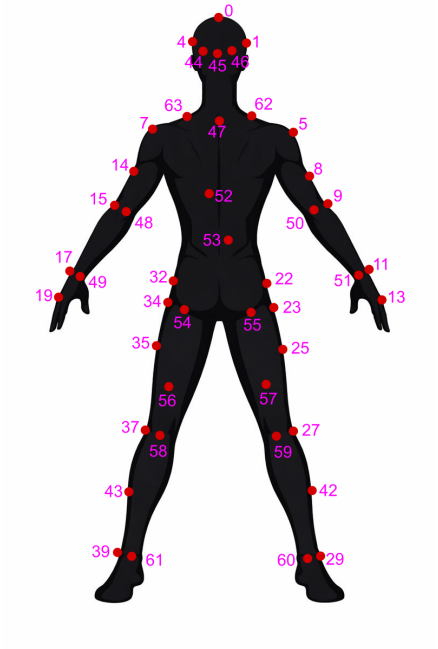
(a) Front view of the body suit.



(b) Back view of the body suit.



(c) Marker locations (front).



(d) Marker locations (back).

Figure 3.3: Motion capture body suit with 64 active LED markers (a,b) and the corresponding marker locations on the abstract model (c,d). The labeling scheme of the markers can be found in Table A.1.

second. While this frame rate is lower than the CSI and motion capture sampling rates, it provides sufficient visual context for qualitative validation and human review of activity execution.

We connect the smartphone to the main host PC via USB cable and run a custom-developed camera application [50] on it. This application enables the host to remotely control video recording with precise timing: the host can command the phone to start and stop recording at an exact specified time, ensuring temporal alignment with the other data modalities. Upon completion of recording, the application automatically transfers the video file from the phone to the main host PC via the USB connection, streamlining the data collection workflow.

We record video at 30 fps to provide visual reference for data inspection and validation. We achieve temporal synchronization with the high-rate CSI and motion capture streams is achieved through a hardware LED-based visual timestamp mechanism, as detailed in Section 3.4.

3.2.5 Network and Orchestration Infrastructure

The multi-modal data capture system requires coordination of multiple hardware components operating at different computers. The infrastructure is organized as follows:

- **Main Host PC:** Serves as the orchestration controller, managing overall capture process and directly interfacing with the ASUS receivers via wired Ethernet connections.
- **USRP Control PC:** Runs a custom application that controls the USRP transmission, enabling the main host to trigger frame transmission remotely.
- **Motion Capture Server:** A dedicated system manages the PhaseSpace motion capture cameras and streams marker position data to the main host.
- **Video Capture Device:** The Samsung S25 smartphone records video with hardware-synchronized LED timing signals.

We synchronize all networked devices using the Network Time Protocol (NTP), with the main host PC serving as the ground truth time reference. This ensures consistent timestamps across all data streams, facilitating subsequent temporal alignment during post-processing.

We developed a custom graphical user interface (GUI) application [51] to configure and coordinate all capture modalities from a single control point. The GUI allows operators to start and stop synchronized recordings across CSI, motion capture, and video systems simultaneously, load predefined activity sequences, and provide real-time instructions to the subject, guiding them through which activities to perform and when to transition between them. This streamlines the data collection workflow, ensures consistent protocol execution, and minimizes human error.

3.2.6 Spatial Configuration

Figure 3.4 presents a top-down view of the recording environment, illustrating all hardware components and defining the subject movement area. The spatial arrangement is carefully designed to balance WiFi signal coverage, motion capture visibility, and realistic activity execution space. Table 3.1 summarizes the key spatial parameters.

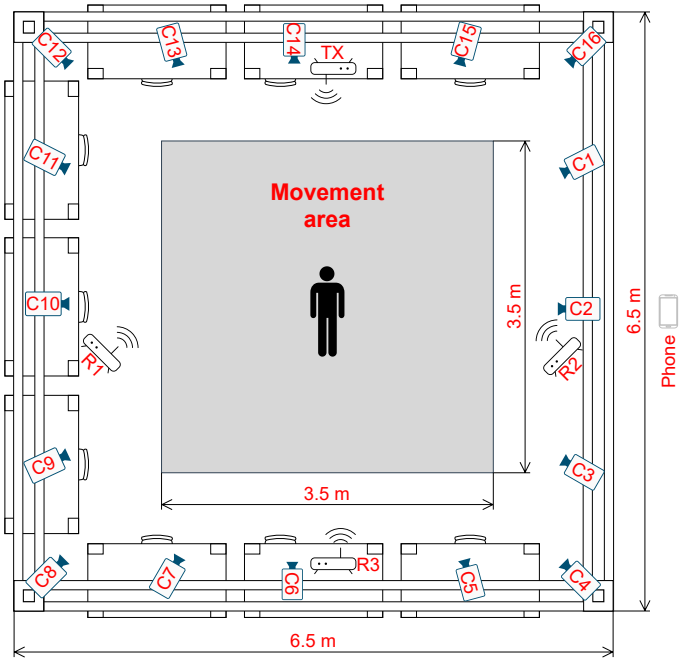


Figure 3.4: Floorplan of the recording environment showing the USRP transmitter (TX), three ASUS receivers (R1, R2, R3), 16 PhaseSpace motion capture cameras (C1–C16), and the 3.5 m × 3.5 m subject movement area (shaded region).

Motion Capture Camera Arrangement As illustrated in Figure 3.5, the cameras are positioned to provide overlapping fields of view converging on the center of the recording environment. This arrangement ensures complete coverage of the subject movement area from multiple viewing angles, minimizing occlusion and maximizing tracking reliability. The PhaseSpace system requires each LED marker to be visible to at least three cameras simultaneously to accurately triangulate its 3D position. The camera placement was optimized during setup to satisfy this constraint throughout the entire movement area.

Video Camera Placement The video camera is positioned at the side of the recording environment, providing a lateral view of the subject during activity execution. The camera placement ensures the synchronization LED is visible in the frame while capturing the full movement area.

Table 3.1: Summary of spatial configuration parameters.

Parameter	**Value**
Recording environment dimensions	6 m × 6 m
Subject movement area	3.5 m × 3.5 m
Transmitter height	1.0 m
Receiver height	1.2 m
Transmitter-to-receiver distance	5.0 m
Receiver spacing	3.0 m
Receiver antenna spacing	2 cm
Motion capture cameras	16
Motion capture camera heights	2.0–3.0 m
Minimum cameras per marker	3

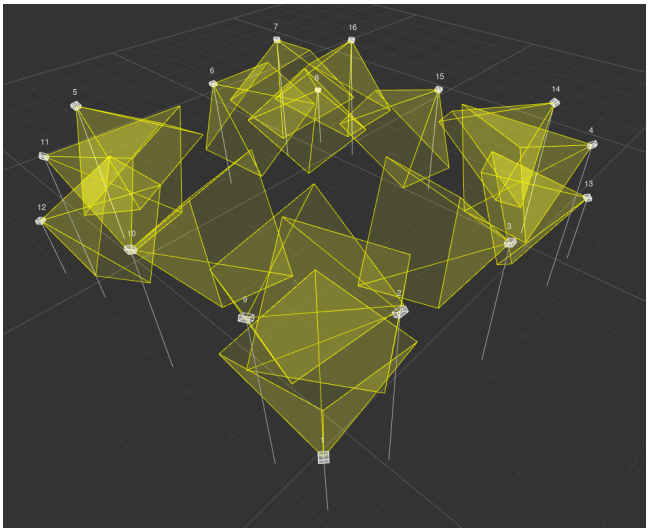


Figure 3.5: PhaseSpace motion capture camera positions and viewing angle.

Spatial Considerations for WiFi Sensing The spatial configuration influences the wireless propagation environment and consequently the observed CSI patterns. The three receivers provide diversity in signal paths: movements closer to one receiver affect its CSI more strongly than distant receivers, enabling spatial localization information to be encoded in the multi-receiver measurements. The 5 m transmitter-receiver separation ensures rich multipath propagation, as signals reflect off walls, furniture, and the subject's body before reaching the receivers. The placement of WiFi devices at torso height (1.0–1.2 m) maximizes sensitivity to human body movements, as this region experiences the greatest propagation path disruption during most activities.

3.3 Data Collection Campaign Design

This section describes the design and execution of the data collection campaign, including the activity vocabulary, randomization strategy, spatial diversity mechanism, and recording workflow.

Choice of Activities The dataset comprises ten distinct activities spanning a range of motion characteristics and complexity levels: *stand, walk, run, squat, jump, jumping jack, raise left arm, raise left foot, stir pot, box.* This activity set includes both coarse-grained full-body motions that produce strong CSI variations across multiple subcarriers, and fine-grained localized movements that generate more subtle signal disturbances, enabling evaluation of segmentation models across varying levels of motion intensity and spatial extent.

Randomization To prevent machine learning models from exploiting artificial patterns or temporal regularities, a custom activity generation script creates randomized activity sequences for each recording session. Each recording session spans approximately 60 seconds, with individual activities assigned random durations between 3 and 10 seconds drawn from a uniform distribution. This duration range balances several considerations: durations shorter than 3 seconds are challenging for subjects to execute consistently and may not provide sufficient samples for model training, while durations longer than 10 seconds would reduce the number of activity transitions per recording, limiting the dataset's utility for segmentation tasks. Activity ordering is randomized across recording sessions, eliminating systematic temporal patterns (e.g., "walk always follows stand") that could be exploited by models. By randomizing both activity order and duration, the dataset reflects realistic variability in human behavior and ensures that segmentation models must rely on CSI signal characteristics rather than dataset-specific biases.

Spatial Diversity and Recording Locations To prevent models from overfitting to location-specific CSI patterns, activities are performed at varied positions within the recording space. Sixteen floor markers are arranged in a grid, and for each recording session, a line of four markers is selected. Dynamic activities are performed along the four-cross line, while static are executed at individual markers within the selected line. This spatial variation introduces diversity in multipath propagation characteristics — different positions relative to transmitter and receiver produce distinct path lengths and reflection patterns — ensuring models must generalize across spatial contexts rather than memorizing location-specific artifacts.

Workflow and Execution Data collection follows a standardized workflow facilitated by a custom graphical user interface (GUI) application that coordinates all capture modalities from a single control point. Each session follows this procedure: (1) motion capture calibration, (2) activity sequence loading into the GUI, (3) subject preparation and positioning at the starting floor marker, (4) pre-recording countdown displaying the first activity and duration, (5) synchronized data capture across CSI, motion capture, and video for approximately

60 seconds, (6) automatic data transfer and storage, and (7) quality verification with re-recording if necessary.

During recording, the GUI provides real-time visual cues to guide the subject: the current activity name and remaining time are displayed, with a preview of the next activity shown at the bottom of the screen. This enables subjects to execute the prescribed sequence accurately without external verbal cues. Subjects immediately begin the next activity when the current one ends, without deliberate pauses. This creates realistic segmentation challenges with ambiguous boundary regions, reflecting the continuous nature of human motion in real-world scenarios.

Activities are performed orthogonal to the transmitter-receiver line-of-sight when possible to maximize multipath effects. For repetitive activities (*jump*, *jumping jack*, *squat*, *raise left arm*, *raise left foot*, *box*, and *stir pot*), subjects perform approximately one repetition per second. For locomotion (*walk* and *run*), subjects maintain a consistent comfortable pace across all sessions.

The protocol intentionally limits variability within individual activity instances to maintain clear activity signatures — excessive intra-activity variability (e.g., changing walking speed mid-activity) would create ambiguous ground truth labels. Instead, variability is introduced through randomized sequences and spatial positioning diversity. This balance ensures the dataset is both challenging and tractable for segmentation model development.

3.4 Data Synchronization and Alignment

Precise temporal alignment across CSI, motion capture, and video data is critical for generating accurate ground truth annotations and enabling multi-modal analysis. We achieve synchronization through a combination of NTP, RTT-based clock offset correction, and a hardware-based visual reference signal.

All networked devices — the USRP control PC, motion capture server, and ASUS receivers — synchronize their clocks to the main host PC using NTP, establishing a common time reference across the data acquisition infrastructure. The ESP32-S3, however, connects to the main host over a UART link rather than the network; for this device we use a custom RTT-based clock offset correction protocol instead. The UART round-trip latency is sub-millisecond, enabling sub-millisecond synchronization accuracy. We re-run this correction immediately before every important command to actively combat clock drift between corrections. Separately, we apply an RTT-based offset correction to synchronize the USRP local clock to its attached PC, ensuring that USRP transmit times are accurately tied to the main host time reference. The video camera does not participate in any of these schemes, necessitating the alternative approach described below.

Hardware Reference Signal To synchronize the video stream, we employ an ESP32-S3 microcontroller running custom firmware [41] with an attached LED positioned within the camera's field of view. As described above, we re-synchronize the ESP32-S3 to the main host via RTT-based correction immediately before each recording to minimize accumulated clock drift. At the precise recording start time, the ESP32-S3 illuminates the LED, creating a visual timestamp that anchors the video stream to the main host time reference.

Coordinated Recording Start At the start of each recording session, the main host PC broadcasts a start command containing a Unix epoch timestamp with microsecond precision. This timestamp specifies the exact future moment when all systems should initiate data capture. The USRP begins transmission, CSI receivers start logging, motion capture begins frame acquisition, and ESP32-S3 triggers the LED — all at this scheduled time. The video camera receives a start command slightly in advance to account for its activation latency and unsynchronized clock, ensuring it is recording when the LED illuminates.

Timestamp Representation Each modality records native timestamps for its data:

- **CSI:** Each CSI measurement nominally includes a timestamp from the ASUS receiver, recorded at packet reception time. However, as noted in Section 3.2, these timestamps are unreliable due to ASUS NTP client inaccuracies and non-deterministic Nexmon-CSI extraction timing. We therefore discard them and instead derive each frame's time by matching its received sequence number to the corresponding USRP transmit time.
- **Motion Capture:** Each frame includes an NTP-synchronized timestamp from the PhaseSpace system, corresponding to the frame capture instant.
- **Video:** The LED turn-on frame is aligned to the recording start time, and subsequent frames are timestamped based on 30 fps frame rate.

All timestamps use Unix epoch time in microseconds, providing a unified temporal representation that enables direct alignment and correlation across modalities during post-processing.

Synchronization Accuracy The combination of NTP for networked devices, RTT-based correction for the ESP32-S3 and USRP, and the hardware LED reference achieves sub-millisecond to millisecond temporal alignment accuracy across modalities. This precision is sufficient for activity segmentation tasks, where activities last 3–10 seconds and temporal features of interest — such as motion onsets and activity transitions — occur on a timescale of tens to hundreds of milliseconds. The synchronization accuracy enables frame-level correspondence between modalities.

3.5 Labeling Methodology

Accurate temporal labels for CSI data are generated using motion capture measurements as ground truth, achieving millisecond-scale annotation precision. A human annotator walks through the recorded sequence and identifies the most appropriate transition points between activities based on the high-rate motion trajectories. The video camera is not actively used in the labeling process, though its recording is displayed in the labeling tool to provide qualitative verification that activities were performed correctly.

3.5.1 Custom Annotation Tool

A custom software application [51] was developed to streamline the labeling process (see Figure 3.6). The tool provides:

- 3D visualization of motion capture marker positions for playback and inspection
- Boundary marking functionality that records exact motion capture frame timestamps
- Activity label assignment for each segment
- Automated CSI label propagation by matching motion capture timestamps to the nearest CSI timestamps (< 1 ms error)

Upon saving annotations, labels are automatically transferred to the CSI data, producing frame-level activity labels ready for model training.

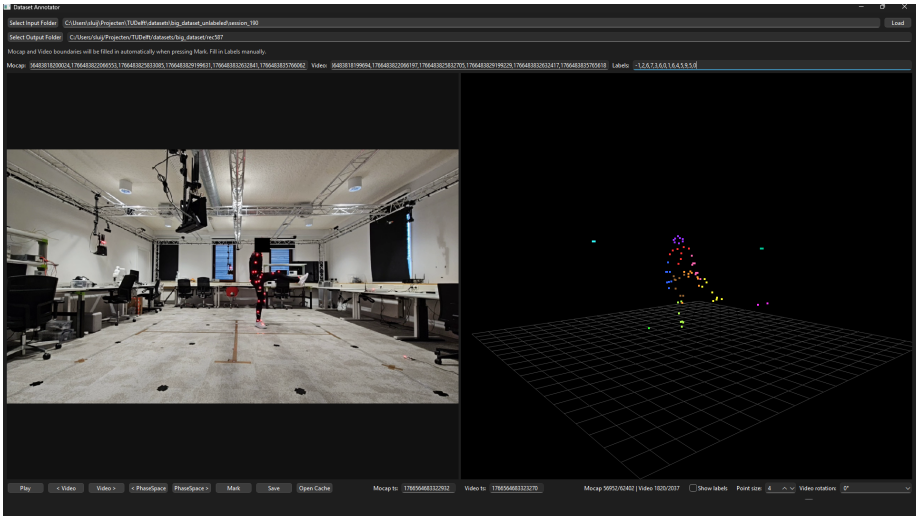


Figure 3.6: Custom annotation tool interface showing 3D motion capture visualization, controls to mark activity boundaries and fill in activity labels. Annotators can play back recordings, place temporal markers at activity boundaries, and assign activity labels, which are automatically propagated to the CSI data upon saving.

3.5.2 Output Format

The annotation process produces:

- Activity boundary timestamps (Unix epoch microseconds)
- Activity labels for each segment
- Frame-level label sequences aligned with CSI measurements (1000 Hz)

After annotation, the data is ready for direct use in training and evaluating segmentation models.

3.6 Dataset Format and Structure

This section describes the file formats and organization of the dataset, facilitating reproducibility and enabling other researchers to utilize the data effectively.

The dataset is organized hierarchically by recording session. Each session directory contains all modalities of a single 60-second recording:

```
dataset/
├── session_01/
│   ├── csi.parquet
│   ├── meta.parquet
│   ├── labels.parquet
│   ├── markers.c3d
│   ├── markers.parquet
│   └── video_phone.mp4
├── session_002/
│   └── ...
```

Table 3.2 summarizes the files in each session directory.

Table 3.2: Files contained in each session directory.

File	Format	Contents
`csi.parquet`	Parquet	1000 Hz CSI measurements (all antennas and subcarriers)
`meta.parquet`	Parquet	Recording parameters (bandwidth, timestamps, frame count)
`labels.parquet`	Parquet	Frame-level activity labels aligned to CSI timestamps
`markers.c3d`	C3D	960 Hz motion capture marker positions (standard format)
`markers.parquet`	Parquet	Same marker data for programmatic access
`video_phone.mp4`	MP4	Synchronized video at 720p, 30 fps (H.264)

3.7 Activity Recognition and Localization Dataset Generation

While the primary purpose of our motion capture system is to provide frame-accurate activity boundaries for temporal segmentation (as described in Section 3.5), the PhaseSpace optical tracking system captures continuous 3D positional data at 960 Hz throughout all recording sessions. This position information enables the generation of additional labels beyond temporal boundaries allowing us to demonstrate the versatility of our data collection infrastructure.

In this section, we describe the generation of a derivative dataset for **joint human activity recognition (HAR) and indoor localization**, following the paradigm introduced by the ARIL model [54]. This derivative dataset assigns each activity window both an activity class label and a discrete floor-plan position label, enabling simultaneous classification of "what" the subject is doing and "where" they are located.

3.7.1 Motivation

The creation of this derivative dataset serves three purposes:

1. **Demonstrating motion capture versatility**: By extracting position labels from the same PhaseSpace recordings used for temporal segmentation, we show that the motion capture system enables multiple applications.

2. **Label quality validation**: Successfully training a localization model provides independent evidence that our motion capture labels are spatially and temporally accurate.

3. **Exploring cross-session generalization**: The ARIL-style task allows us to investigate how well WiFi CSI features generalize across different recording sessions; a key challenge for real-world deployment.

3.7.2 Floor-Plan Grid Discretization

The recording space is divided into a 4×4 grid of 16 discrete floor positions, Figure 3.7 shows the layout.

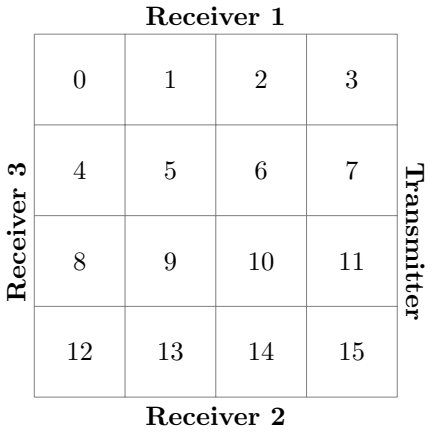


Figure 3.7: Floor-plan grid discretization. The recording space is divided into 16 positions.

Each recording session covers a single row or column of the grid (the subject walks along one line and performs activities at four positions), rather than all 16 positions. This constraint is enforced during label assignment to achieve a balanced usage of each position, and it prevents cross-line misclassifications caused by coordinate drift between sessions.

3.7.3 Position Label Generation Pipeline

Generating consistent position labels from PhaseSpace motion data presents several challenges:

- **Marker dropout**: Individual markers are occasionally occluded, producing missing data.

- **Coordinate drift**: The PhaseSpace system requires periodic recalibration between recording sessions, potentially shifting the absolute XZ coordinates.
- **Per-recording quantization errors**: Noise in the averaged position can cause windows to be quantized to adjacent grid cells.

We address these challenges through a multi-stage pipeline:

Stage 1: Marker Selection and Preprocessing

For each activity window, we compute the subject's floor-plan position from a body-center marker set comprising 16 markers on the torso, back, neck, shoulders, and hips. Arms and legs are excluded to ensure the position estimate reflects center-of-mass rather than gesture or footstep locations.

The following preprocessing steps are applied per marker per window:

1. **Dropout detection**: Markers producing all-zero coordinates in $> 10\%$ of frames are excluded. At least 3 markers are always retained.
2. **Gap interpolation**: Missing data runs (≤ 960 frames per second) are linearly interpolated. Longer gaps are left as NaN.
3. **Outlier removal**: Values exceeding $5 \times \text{IQR}$ from the median on any axis are replaced via interpolation.
4. **Per-frame averaging**: Valid markers are averaged in 3D per frame; frames with fewer than 2 valid markers are discarded.
5. **Median smoothing**: A kernel-5 median filter suppresses residual per-frame jumps.
6. **Floor-plane projection**: The Y-axis (height) is discarded; only X and Z coordinates are retained.

Stage 2: Per-Line Calibration

The recordings contain T-pose activities at the start, and for a subset of these recordings the grid position of the T-pose activity is manually annotated for calibration. For each row or column, we extract the median XZ positions observed during T-pose windows and map them to the known grid nodes, establishing a spatial calibration for that line. This calibration is reused for all other recordings on the same line.

Stage 3: Initial Label Assignment

For each activity window in each recording:

1. Compute the median torso XZ position across all frames in the window.
2. Quantize the position to one of the 4 grid cells on the recording's annotated row or column using the line calibration.
3. Store the discrete position label (0–15) and the raw XZ coordinates.

Walking and running activities are excluded because subjects traverse multiple grid cells during these activities, making it impossible to assign a single position label. The T-Pose activity is also excluded, as this was performed for calibration purposes, not the actual dataset.

Stage 4: Global KMeans Correction

After all recordings are processed, residual quantization errors remain due to noise and drift. We apply a global correction:

1. Collect all raw XZ coordinates (x, z) across all windows from all recordings.
2. Fit KMeans with $k = 16$ clusters.
3. Relabel each window with the index of its nearest cluster centroid.
4. Sort cluster centroids into row-major order.

This step step corrects $\sim$67% of the 5,749 windows in the full dataset, significantly improving spatial consistency.

Figure 3.8 shows the final distribution of windows in XZ space, colored by corrected position label.

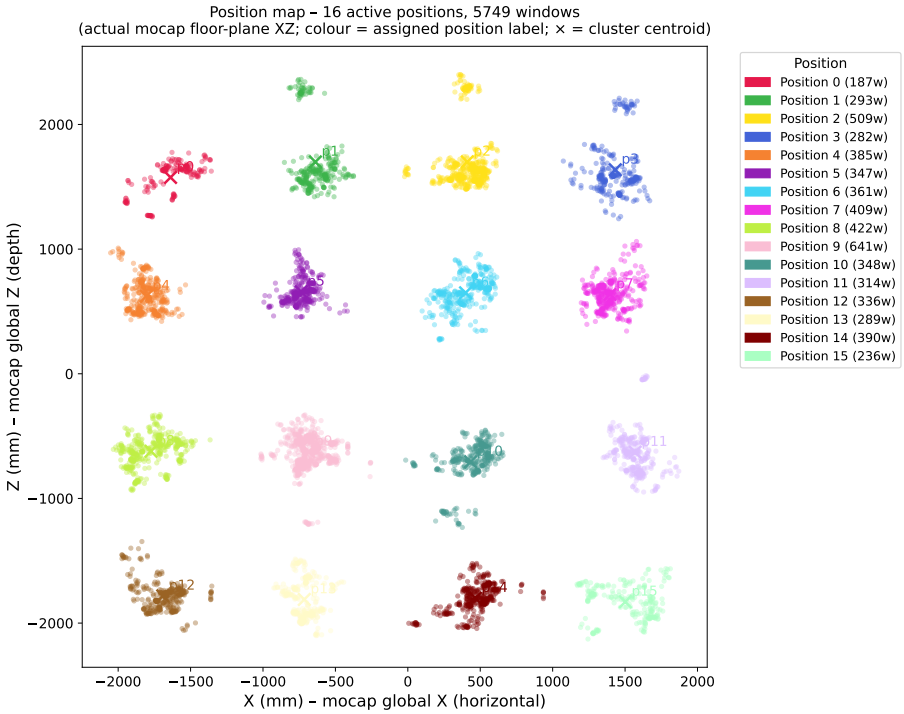


Figure 3.8: Scatter plot of all activity windows in floor-plane (XZ) coordinates, colored by final position label after global KMeans correction. The 16 clusters correspond to the grid cells in Figure 3.7.

3.7.4 Dataset Statistics

The resulting activity recognition and localization dataset comprises 5,479 activity windows extracted from 110 recording sessions. Table 3.3 summarizes the key properties.

Table 3.3: Activity recognition and localization dataset statistics

Property	**Value**
Total windows	5,749
Recording sessions	110
CSI input shape	$(224, 700)$
	4 antennas $\times$ 56 subcarriers $\times$ 700 time-steps
Activity classes	8
Position classes	16
Activities excluded	Walking, Running

Each window is represented by:

- **CSI amplitude**: Shape $(224, 700)$, derived from 4 antennas $\times$ 56 subcarriers sampled at 700 time-steps per window.
- **Activity label**: Integer in range 0–7.
- **Position label**: Integer in range 0–15 corresponding to the floor-plan grid cell.
- **Session ID**: Recording session number, used for cross-session evaluation splits.

3.7.5 Summary

By leveraging the continuous 3D tracking capabilities of our PhaseSpace motion capture system, we have generated a derivative dataset suitable for joint activity recognition and indoor localization. The multi-stage position labeling pipeline addresses marker dropout, coordinate drift, and quantization errors through preprocessing, per-line calibration, and global clustering correction. This demonstrates that our data collection infrastructure supports multiple downstream tasks beyond the temporal segmentation problem. Experimental results using this dataset are presented in Section 6.3.

3.8 Doppler Temporal Segmentation Datasets

The WiPos dataset provides continuous, motion-capture-labeled recordings with raw CSI amplitude at 1000 Hz across three receivers. Section 3.7 derived a windowed HAR and localization dataset from these recordings, representing each fixed-length activity window as a raw CSI amplitude snapshot with a single activity label and a discrete floor-plan position label

We construct a different kind of derivative dataset, targeting temporal activity segmentation. Two differences set it apart. First, the task requires the full

continuous label stream rather than isolated window classifications. Second, raw CSI amplitude is position-dependent: it encodes the static multipath channel structure, which shifts as the subject moves to different locations across recording sessions. To reduce this sensitivity, the datasets described here replace raw amplitude with Doppler features derived from the temporal conjugate product of consecutive complex CSI samples. Because Doppler features are computed from the difference between consecutive samples, they suppress the static channel contributions (LOS path, wall reflections) that dominate the raw channel and are strongly position-dependent. The remaining dynamic component reflects changes in the active reflection paths, which depend primarily on how body parts move relative to the receiver geometry rather than on absolute position. At a constant orientation with respect to the LOS axis, motion-induced path changes are therefore less sensitive to where in the room the subject stands: a change in subject orientation alters the dominant reflection geometry and shifts the feature distribution. The practical benefit is that recurring motion patterns, such as cyclical gestures, produce more consistent Doppler signatures across locations than raw amplitude, which undergoes near-noise-level fading as the static channel shifts with position. These Doppler datasets serve as the training and evaluation data for DopplerTAS (Chapter 5), and are provided in two variants derived from the same WiPos recordings: a three-receiver (3RX) dataset and a one-receiver (1RX) subset used for the ablation study in Section 6.2.3.

3.8.1 Doppler Feature Extraction

Physical motivation. As described in Section 2.1, the received signal on subcarrier k is a superposition of contributions from multiple propagation paths (see Eq. 2.2) and the measured phase (see Eq. 2.3). Each path i contributes a phase term $\phi_{i,k}(t) \propto \tau_i(t)$, where $\tau_i(t)$ is the propagation delay of that path. When a moving object shifts the reflector along path i, the delay $\tau_i(t)$ changes over time, and the corresponding phase $\phi_{i,k}(t)$ increases or decreases continuously.

A phase that increases linearly with time is equivalent to a constant frequency offset. In the context of wave propagation, such a constant frequency offset is mathematically equivalent to a Doppler shift — even though physically it arises from the superposition of many scattered paths rather than from a single point reflector moving at a well-defined radial velocity. Because every moving body part contributes its own path, the resulting motion-induced spectrum is a broadened distribution, not a single neat spectral line.

Step 1 — AGC gain removal. As discussed in Section 2.1, automatic gain control (AGC) introduces large-scale amplitude jumps between packets. For Doppler computation specifically, these jumps produce discontinuities in the CSI sequence that manifest as high-frequency artifacts in the subsequent STFT. Each CSI frame is therefore first normalized by its L1 norm across subcarriers:

$$\hat{H}_k = \frac{H_k}{\|\mathbf{H}\|_1} = \frac{H_k}{\sum_{m=1}^{K} |H_m|}, \tag{3.1}$$

where K is the number of subcarriers. Dividing by the L1 norm removes the large-scale gain common to all subcarriers in a packet while preserving the relative per-subcarrier structure.

Step 2 — Antenna conjugation. Per-packet hardware timing errors (e.g. STO) and other front-end phase offsets are shared across all subcarriers within a packet but are randomized between packets (Section 2.1). Multiplying each non-reference antenna by the complex conjugate of a reference antenna cancels this common per-packet phase:

$$\tilde{H}_{t,a} = \hat{H}_{t,a} \odot \overline{\hat{H}_{t,0}}, \qquad a = 1, 2, 3, \tag{3.2}$$

where antenna 0 of each receiver server as the phase reference. This operation reduces each receiver's four physical antennas to three differential channels. The antenna conjugation is applied independently per receiver.

Step 3 — Temporal conjugate product. To suppress the static multipath channel (LOS path, wall reflections) and isolate motion-induced dynamics, we compute the temporal conjugate product between consecutive frames [32]:

$$\mathbf{R}_t = \mathbf{H}_t \odot \overline{\mathbf{H}_{t-1}}, \tag{3.3}$$

where the product is taken independently over all antenna channels and subcarrier indices. Writing each element in polar form, $H_t = |H_t|\, e^{j\phi_t}$, the conjugate product becomes

$$R_t = |H_t| \cdot |H_{t-1}|\, e^{j(\phi_t - \phi_{t-1})}. \tag{3.4}$$

The differential phase $\Delta\phi_t = \phi_t - \phi_{t-1}$ retains only components that change between consecutive samples: contributions that are constant across frames — the dominant static multipath — cancel out. The remaining signal reflects motion-induced path-length changes and therefore carries the Doppler content.

Step 4 — Short-time Fourier transform. The time series $\{R_t\}$ of conjugate products is processed with a short-time Fourier transform (STFT) to obtain time-resolved local frequency content: the STFT slides a fixed-length analysis window along $\{R_t\}$ and computes the spectrum within each window, yielding one power spectrum per step. For Doppler frame m centered at raw-sample position $s = mH$, the STFT is

$$\tilde{D}_{m,k} = \sum_{n=0}^{W-1} R_{s+n}\, w[n]\, e^{-j2\pi kn/W}, \qquad k = 0, \ldots, W-1, \tag{3.5}$$

where $W = 512$ is the window length (approximately 0.5 s at the 1,000 Hz raw CSI rate), $H = 10$ is the hop size between consecutive Doppler frames (0.01 s; chosen primarily to reduce computational overhead), and $w[\cdot]$ is a rectangular window. The Doppler power at frame m and frequency bin k is obtained by averaging the squared magnitude over the $C = 30$ subcarriers:

$$D_{m,k} = \frac{1}{C} \sum_{c=1}^{C} \left|\tilde{D}_{m,k,c}\right|^2. \tag{3.6}$$

Averaging over subcarriers improves the signal-to-noise ratio by exploiting frequency diversity: different subcarriers experience independent small-scale fading but carry the same Doppler signature, so their mean is a more stable estimate of the underlying motion content. The power spectrum D_m is then normalised to $[0, 1]$ by dividing by its maximum value, converted to decibels, and clipped at -30 dB.

Frequency range. Of the $W = 512$ FFT bins, $F = 61$ bins are retained (bins 0 through 60). At a CSI rate of 1000 Hz the frequency resolution is $\Delta f = 1000/512 \approx 1.95$ Hz per bin, so the retained range spans approximately $[0, 117]$ Hz. Bin 0 is the DC component, corresponding to zero Doppler shift, which captures the static background and distinguishes motionless periods from active motion. The upper limit of ≈117 Hz is chosen empirically to cover the motion-induced phase variation rates produced by human activities at typical movement speeds; higher frequencies do not appear to contain discriminative activity content in the recordings.

Since the input $\{R_t\}$ is complex-valued, the W-point FFT produces a two-sided, asymmetric spectrum: bins 0 to $W/2 - 1$ correspond to positive frequencies and bins $W/2$ to $W - 1$ correspond to negative frequencies. We retain $F = 61$ bins (bins 0 to 60), covering the positive-frequency range $[0,\ 60 \cdot \Delta f] \approx [0,\ 117]$ Hz at a frequency resolution of $\delta F = 1000/512 \approx 1.95$ Hz per bin. Bin 0 is the DC component corresponding to zero Doppler shift, which provides a discriminative baseline distinguishing motionless from active periods. In practice, human activity Doppler signatures at typical movement speeds are generally concentrated below ≈ 60 Hz; the retained range is therefore wide enough to capture the relevant content.

3.8.2 Dataset Construction

3RX dataset. The 3RX dataset uses all three ASUS receivers deployed in the recording space. For each receiver, the first three antennas are retained and averaged over the subcarrier-stream dimension, yielding a complex time series per antenna. Stacking three receivers, each contributing three antenna series, gives $3 \times 3 = 9$ independent complex time series. The temporal conjugate product and STFT are applied independently to each of these nine channels, producing nine simultaneous Doppler power spectra. The per-frame feature is therefore a set of nine independent velocity-projection measurements:

$$\mathbf{X}_t \in \mathbb{R}^{9 \times 61}, \quad 9 = 3 \text{ receivers} \times 3 \text{ antennas per receiver.}$$

1RX subset. For the spatial-diversity ablation in Section 6.2.3, a 1RX dataset variant is produced by retaining only the three channels of one receiver from the 3RX features, giving $\mathbf{X}_t \in \mathbb{R}^{3 \times 61}$ per frame. No separate recording or feature extraction pass is required; the 1RX data is a strict subset of the 3RX features.

3.8.3 Temporal Label Preservation

Each Doppler frame must be assigned an activity label. Because the STFT window for frame m spans raw CSI samples $[mH, mH + W)$, its representative timestamp is the sample at the window center, $mH + W/2$. The frame receives the activity label by identifying which motion-capture annotated activity interval the representative timestamp falls within.

Each recording is saved as a triplet (`doppler.npy`, `labels.npy`, `meta.json`). `doppler.npy` holds the complete sequence of Doppler frames for that recording as a continuous array of shape $(T, 9, 1, 61, 1)$, where T is the number of STFT hops (one per 0.01 s of recording); `labels.npy` holds the corresponding integer class label for every frame; `meta.json` records the recording identifier, total

frame count, and per-class frame counts. A `dataset_info.json` aggregates statistics across all recordings.

3.8.4 Dataset Summary

Table 3.4: Doppler temporal segmentation dataset statistics (3RX).

Property	**Value**
Recordings processed	209
Total labeled frames	$\approx$1.30 M
Background frames (class 0)	$\sim$4–5%
Feature shape per receiver	$(3, 1, 61, 1)$
Number of receivers	3
Number of classes	11
STFT window W / hop H	512 / 10
Window / hop duration	$\approx$0.5 s / 0.01 s
Frequency bins F	61
Frequency range	$[0, 117]$ Hz

Chapter 4

Benchmarking Framework

An obstacle in WiFi sensing research is that datasets and models are published with incompatible formats and evaluation protocols, making fair and reproducible comparisons across methods difficult. To address this, we developed a unified benchmarking framework that provides a single environment for training, evaluating, and comparing WiFi CSI-based activity segmentation models across multiple datasets. We call this framework **Breaking-CSI** [37]. Section 4.1 motivates the framework and states its design goals. Section 4.2 describes the software architecture. Section B explains how datasets and models are integrated. Section 4.3 defines the standardized evaluation protocol and metrics.

4.1 Design Goals

The development of WiFi sensing research has produced numerous datasets and machine learning models, each published with different data formats, preprocessing pipelines, and interface requirements. This fragmentation creates significant barriers to reproducible research and fair model comparison. This section outlines the design goals that motivated the development of the Breaking-CSI benchmarking framework.

4.1.1 Addressing the Fragmentation Problem

A fundamental challenge in WiFi sensing research is the lack of standardization across published work. Each research group releases datasets in different file formats with varying data shapes, sampling rates, and preprocessing approaches. Similarly, each published machine learning model expects specific input dimensions and data organizations that rarely align with other models or datasets. This creates several critical problems:

- **Limited Model Evaluation:** Researchers cannot easily test their models on multiple datasets without significant effort to reformat and preprocess each dataset differently.
- **Unfair Comparisons:** Comparing models across papers is difficult or impossible because different preprocessing pipelines, data splits, and evaluation protocols produce incomparable results.

The Breaking-CSI framework addresses these issues by providing a unified platform where datasets and models can be integrated once and then freely combined for evaluation.

4.1.2 Core Design Principles

The framework is guided by four core design principles:

Standardization All datasets are converted to a common data format and shape, and all models are adapted to accept this standardized input. This eliminates the need for per-dataset or per-model data reshaping, enabling any model to run on any supported dataset without modification.

Reproducibility The framework enforces deterministic behavior through fixed random seeds and provides comprehensive logging of all experimental runs. Model checkpoints are saved automatically, and all experimental configurations are explicitly defined, ensuring that results can be reliably reproduced.

Fairness All datasets are processed through the same preprocessing pipeline with configurable normalization and augmentation options. Evaluation protocols, data splits, and metrics are standardized across experiments, ensuring that model comparisons reflect true performance differences rather than artifacts of different evaluation setups.

Extensibility The framework is not limited to the datasets and models currently supported. New datasets and models can be added through base classes and integration procedures, allowing the framework to accommodate future research contributions.

4.2 Framework Architecture

This section describes the technical architecture of the Breaking-CSI framework, detailing how the design principles outlined in Section 4.1 are implemented through modular components. Figure 4.1 provides an overview of the framework's architecture and data flow.

4.2.1 Dataset Loader Interface

Published WiFi sensing datasets exist in diverse file formats including Parquet, NumPy, MATLAB, with each dataset employing different data organizations such as nested lists, flat arrays, or multi-dimensional tensors. The dataset loaders interface handles this through a two-stage process:

1. **Initial Conversion:** A dataset-specific loader reads the original data format and converts it into the framework's common representation: NumPy arrays with shape `(timestamp, antennas, streams, subcarriers, 2)`, where the final dimension contains amplitude and phase components. This conversion is performed once per dataset.

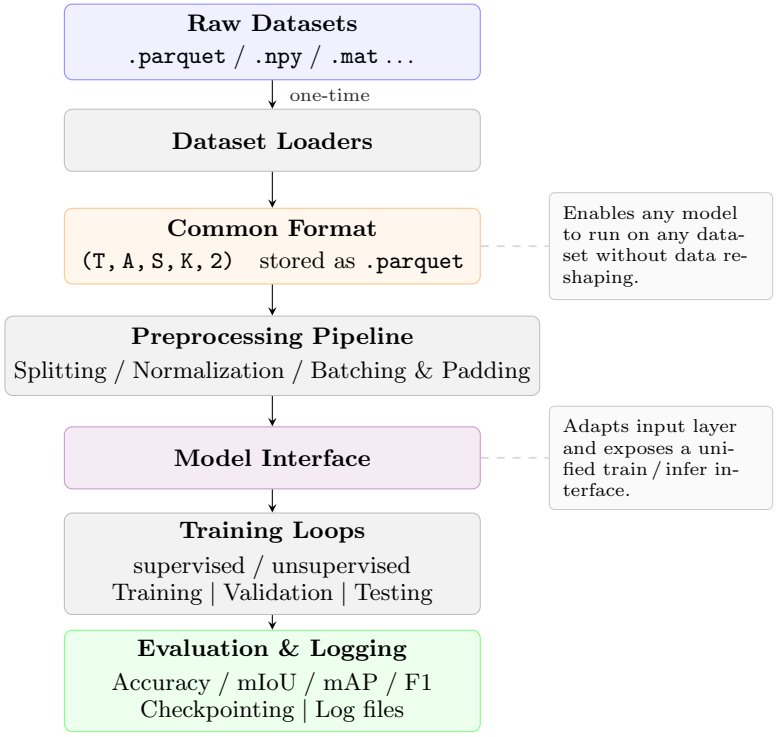


Figure 4.1: Breaking-CSI framework architecture. Raw datasets in various formats are converted once by dataset-specific loaders into a common `(T, A, S, K, 2)` representation — where T is the number of time steps, A the number of antennas, S the number of streams, K the number of subcarriers, and the final dimension of size 2 contains amplitude and phase — stored as Parquet files, enabling any model to run on any dataset without further reshaping. The preprocessing pipeline applies splitting, normalization, and batching. Models are integrated via a common interface and trained with standardized supervised or unsupervised loops. Evaluation produces consistent metrics for fair comparison across models and datasets.

2. **Efficient Storage:** The converted dataset is stored in Parquet format, an efficient columnar storage format that enables fast loading and compression. Subsequent uses of the dataset load directly from the Parquet files, avoiding redundant conversion overhead.

This approach ensures that the costly conversion process occurs only once, while subsequent experiments benefit from fast, standardized data loading.

4.2.2 Preprocessing Pipeline

The preprocessing pipeline applies a sequence of transformations to prepare data for model training and evaluation. All datasets pass through the same pipeline, with configurable parameters to accommodate different experimental requirements:

Data Splitting The dataset is partitioned into training, validation, and test sets according to user-specified split ratios.

Normalization Each split is normalized independently to prevent data leakage from training to test sets. The framework supports multiple normalization strategies including AGC, Z-score and Min-Max normalization.

Batching and Collation Samples are grouped into batches for efficient processing. Since CSI sequences may have varying lengths, the collation function applies padding to match the longest sequence in each batch, enabling batched tensor operations while preserving temporal information.

4.2.3 Model Interface

Published models are typically designed for specific dataset formats and input shapes. To integrate a model into the framework, it must be adapted to accept the common data format: `(timestamp, antennas, streams, subcarriers, 2)`. This adaptation involves modifying the model's input layer or adding a transformation layer that reshapes the standardized input into the model's expected format internally.

Once adapted, the model implements a common interface that exposes standardized methods for training, validation, and inference. This enables the framework to treat all models uniformly regardless of their internal architecture.

Procedures for integrating new datasets and models into the framework are described in Appendix B.

4.2.4 Training and Evaluation Loops

Supervised Training Loop For supervised models, the training loop executes the following workflow:

1. Train for N epochs on the training set, performing backpropagation and weight updates
2. Evaluate on the validation set after each epoch
3. After N epochs, evaluate the model on the test set

During evaluation, the framework computes and reports multiple metrics including accuracy, mean Intersection over Union (mIoU), mean Average Precision (mAP), and F1 score, providing comprehensive performance assessment.

Unsupervised Training Loop For unsupervised models that do not require iterative training, the framework provides a simplified loop that applies the model directly to the data and evaluates the results using the same metrics as supervised models, enabling fair comparison.

4.2.5 Logging and Checkpointing

The framework automatically logs all experimental runs, recording:

- Experiment configuration (model, dataset, hyperparameters, preprocessing settings)
- Training progress (loss, validation metrics per epoch)
- Final evaluation metrics on the test set

Two types of model checkpoints are saved: one checkpoint is saved after every epoch, overwriting the previous one, and one checkpoint is saved only when the validation loss improves. All logs and checkpoints are organized by experiment timestamp and configuration, supporting reproducibility and result tracking.

4.2.6 Visualization Hooks

Beyond scalar metrics and checkpoints, the framework exposes two hook interfaces that allow custom visualization logic to be attached without modifying the training loops.

Prediction hooks A prediction hook receives the raw predictions and ground-truth labels at the end of every evaluation batch and epoch. Two built-in implementations are provided:

- **Timeline hook:** Produces a two-row color-bar plot per epoch, with ground-truth labels in the top row and model predictions in the bottom row for a configurable number of representative samples. This makes it easy to inspect whether the model correctly detects segment boundaries and which activities are confused at transition points.
- **Confusion matrix hook:** Accumulates predictions across all batches and writes a per-class confusion heat-map image at epoch end, revealing misclassifications and class collapse.

Embedding hooks An embedding hook receives intermediate feature tensors (embeddings) produced by the model during inference, enabling analysis of the learned representation space. The built-in t-SNE hook collects up to a configurable number of embedding vectors per class, optionally reduces dimensionality with PCA, and then generates a t-SNE scatter plot. Color-coding by class label makes it straightforward to assess whether the model has learned class-separable representations.

Both hook types follow the same factory pattern — a configuration object is declared in the experiment YAML, and the framework instantiates and registers the corresponding hook before the evaluation loop begins. Additional hooks implementing the prediction hook or embedding hook protocols can be added to extend the framework with custom analyses without modifying the training loop code.

4.3 Evaluation Protocol

The Breaking-CSI framework employs a standardized evaluation protocol to ensure fair and reproducible model comparisons. This section describes the metrics used for performance assessment and reproducibility considerations.

4.3.1 Evaluation Metrics

The framework computes four complementary metrics to provide comprehensive performance assessment for activity segmentation and recognition:

Accuracy Frame-level classification accuracy measures the proportion of correctly classified frames across all test samples:

$$\text{Accuracy} = \frac{\text{Number of correctly classified frames}}{\text{Total number of frames}} \tag{4.1}$$

While intuitive, accuracy can be misleading for imbalanced datasets where some activities are much more frequent than others.

Mean Intersection over Union (mIoU) IoU measures the overlap between predicted and ground truth segments for each activity class. For a given class c, IoU is computed as:

$$\text{IoU}_c = \frac{TP_c}{TP_c + FP_c + FN_c} \tag{4.2}$$

Where TP_c is true positives, FP_c is false positives, and FN_c is false negatives. The mean IoU averages across all classes:

$$\text{mIoU} = \frac{1}{C} \sum_{c=1}^{C} \text{IoU}_c \tag{4.3}$$

Where C is the number of activity classes. mIoU is particularly effective for segmentation tasks as it penalizes both over- and under-segmentation.

Mean Average Precision (mAP) Average Precision (AP) summarizes the precision-recall curve for each class by computing the area under the curve. Mean Average Precision averages AP across all classes:

$$\text{mAP} = \frac{1}{C} \sum_{c=1}^{C} \text{AP}_c \tag{4.4}$$

mAP is robust to class imbalance and provides insight into model performance across varying confidence thresholds.

F1 Score The F1 score is the harmonic mean of precision and recall, computed per-class and then averaged:

$$\text{F1}_c = 2 \cdot \frac{\text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c} \tag{4.5}$$

$$\text{F1} = \frac{1}{C} \sum_{c=1}^{C} \text{F1}_c \tag{4.6}$$

F1 provides a balanced measure that accounts for both false positives and false negatives.

By reporting all four metrics, the framework enables comprehensive assessment of model performance from multiple perspectives, reducing the risk of overlooking important performance characteristics.

4.3.2 Reproducibility Considerations

To ensure reproducibility results, the framework enforces several practices:

- **Fixed random seeds:** All random number generators (NumPy, PyTorch, Python random module) are seeded with a user-specified value, ensuring deterministic behavior across runs.
- **Explicit configurations:** All hyperparameters, preprocessing settings, and evaluation parameters are specified in configuration files that are logged alongside results.
- **Comprehensive logging:** All experimental details — including dataset, model, training duration, and hyperparameters — are logged, enabling precise reproduction of results.

These practices, combined with the standardized preprocessing and evaluation pipelines, ensure that results produced by the framework are comparable, verifiable, and reproducible.

Chapter 5

DopplerTAS Model

This chapter introduces DopplerTAS [37], a model for WiFi-based temporal activity segmentation designed to generalize across the spatially varied recordings in the WiPos dataset. Instead of raw CSI amplitude, DopplerTAS operates on Doppler features derived from the time-differential channel phase, which are largely position-invariant. Section 3.8 (in Chapter 3) describes the Doppler feature extraction and the temporal segmentation dataset variant used for training.

5.1 The DopplerTAS Model

5.1.1 Motivation

The temporal segmentation task requires assigning an activity label to every frame in a continuous recording. This differs from activity recognition, where a pre-segmented clip receives a single label; here the model must implicitly locate activity boundaries while classifying the activity within each segment. The core difficulty is that accurate boundary placement requires temporal context on both sides of a transition: a frame near the onset of an action is ambiguous without the subsequent motion confirming the activity, and an offset frame is ambiguous without prior context establishing what just completed.

Why Doppler features instead of raw CSI. Existing WiFi temporal segmentation and recognition models process raw CSI amplitude [71, 34]. Raw CSI amplitude encodes the full multipath channel, which is determined jointly by the subject's motion and their position relative to the antenna array. Because signal paths change when the subject stands at a different location in the room, even the same activity performed at different positions produces a different CSI pattern. In a dataset where subjects move freely throughout the recording space — as in the WiPos dataset — this is an instance of covariate shift: the input distribution seen during training differs from the distribution encountered during testing, even though the activity labels are the same. A model trained on recordings from one set of positions may therefore not generalize reliably to recordings from different positions. The benchmarking results presented in Section 6.1 confirm this: raw-CSI models trained on the recordings

show substantially lower accuracy than on public datasets where subjects are constrained to a fixed position.

Doppler features, by contrast, capture the rate of change of the channel phase rather than the raw channel itself. As derived in Section 3.8.1, the temporal conjugate product suppresses static multipath contributions and retains only motion-induced phase dynamics. However, the Doppler spectrum is not fully position-independent. The shift in a reflection path caused by a limb movement depends on the angle between the limb, the transmitter, and the receiver: at different room positions, the same motion produces slightly different path-length changes, and therefore slightly different Doppler shifts. Orientation has a similar effect: turning the subject changes which body parts move toward which receivers, altering the per-receiver velocity projections. What changes less with position is the overall spectral shape for a given motion type, because the dominant contribution comes from how fast body parts are moving, not from the exact geometry. For periodic or recurring motions — such as squatting, jumping, or stirring — this means the motion-induced spectral pattern repeats more consistently across locations than raw amplitude, which is dominated by the static channel and undergoes near-noise-level fading with position. In the Wi-Pos dataset, subject orientation is controlled (activities are performed facing a fixed direction), which further reduces orientation-induced variability. Doppler features are therefore expected to generalize better than raw amplitude across the spatially varied recordings; we test this hypothesis directly in the evaluation.

Architecture overview. DopplerTAS addresses the segmentation task with a three-stage fully supervised architecture. A linear input embedding first projects each Doppler frame to a compact representation. A multi-layer bidirectional long short-term memory (BiLSTM) network then integrates temporal context across the full input window simultaneously in the forward and backward directions, giving every frame access to both past and future evidence. Finally, a lightweight per-frame classifier maps each contextualized representation to class logits, producing one prediction per input frame.

The model is trained end-to-end with weighted cross-entropy loss and requires no attention mechanism, convolutional backbone, or pretraining.

Multi-receiver input fusion. Fusion of the receiver streams is performed at dataset generation time (Section 3.8.2). Each receiver's Doppler channels are extracted independently and then concatenated along the channel axis, giving 9 independent input channels for the 3RX configuration. The model therefore treats all 9 channels as a flat set of parallel time series; it has no explicit knowledge of which channels originate from which receiver. Alignment across receivers is by frame count: the middle receiver serves as the temporal reference — its frame count and timestamps define the common timeline — and the other receivers are truncated or zero-padded to match. No per-packet cross-receiver sequence-number matching is performed. As a result, individual packet loss at one receiver affects only that receiver's channels; the other channels remain unaffected.

5.1.2 Architecture Overview

The model accepts a window of T consecutive Doppler frames as input. Each frame is represented as a vector $\mathbf{x}_t \in \mathbb{R}^{d_{\text{in}}}$ obtained by flattening the per-frame feature map of shape $A \times 1 \times F \times 1$, giving $d_{\text{in}} = A \cdot F$, where A is the number of independent Doppler time series and $F = 61$ is the number of frequency bins retained from the STFT. The architecture is generic in A; in the 3RX configuration $A = 9$ (three receivers, each with three antennas) and in the 1RX configuration $A = 3$ (one receiver), giving $d_{\text{in}} = 549$ and $d_{\text{in}} = 183$, respectively.

Stage 1 — Input Embedding. Each frame $\mathbf{X}_t$ is independently projected to a d_e-dimensional embedding space by a linear layer followed by ReLU activation and dropout:

$$\mathbf{e}_t = \text{Dropout}\big(\text{ReLU}(\mathbf{W}_e\,\mathbf{x}_t + \mathbf{b}_e)\big), \quad \mathbf{e}_t \in \mathbb{R}^{d_e}, \tag{5.1}$$

where $\mathbf{W}_e \in \mathbb{R}^{d_e \times d_{\text{in}}}$ and $\mathbf{b}_e \in \mathbb{R}^{d_e}$ are learned parameters, $d_e = 512$, and dropout probability $p = 0.3$. The same linear projection is applied independently to every frame, so no temporal information is introduced at this stage.

Stage 2 — Bidirectional LSTM. The embedding sequence $(\mathbf{e}_1, \ldots, \mathbf{e}_T)$ is processed by a three-layer bidirectional LSTM [22, 47] with hidden dimension $h = 512$ per direction:

$$\overleftrightarrow{\mathbf{h}}_t = \text{BiLSTM}\big(\mathbf{e}_1, \ldots, \mathbf{e}_T\big), \quad \overleftrightarrow{\mathbf{h}}_t \in \mathbb{R}^{2h}, \tag{5.2}$$

where $\overleftrightarrow{\mathbf{h}}_t$ is the concatenation of the forward hidden state $\overrightarrow{\mathbf{h}}_t \in \mathbb{R}^h$ and the backward hidden state $\overleftarrow{\mathbf{h}}_t \in \mathbb{R}^h$. Bidirectionality is motivated by the asymmetry of activity boundaries: a frame at the onset of an action carries little discriminative information on its own — it needs subsequent frames to confirm the activity. The backward pass allows the model to propagate this future evidence back to earlier frames, directly helping with onset localization. Depth (three stacked layers) increases the capacity to represent non-linear temporal dynamics. Inter-layer dropout is applied between layers.

Stage 3 — Per-Frame Classifier Each contextualized representation $\overleftrightarrow{\mathbf{h}}_t$ is independently mapped to class logits by a two-layer MLP with GELU activation [21]:

$$\hat{\mathbf{y}}_t = \mathbf{W}_2\,\text{GELU}\big(\text{Dropout}(\mathbf{W}_1\,\overleftrightarrow{\mathbf{h}}_t + \mathbf{b}_1)\big) + \mathbf{b}_2, \quad \hat{\mathbf{y}}_t \in \mathbb{R}^C, \tag{5.3}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_{\text{mlp}} \times 2h}$, $\mathbf{b}_1 \in \mathbb{R}^{d_{\text{mlp}}}$, $\mathbf{W}_2 \in \mathbb{R}^{C \times d_{\text{mlp}}}$, and $\mathbf{b}_2 \in \mathbb{R}^C$ are learned parameters, with MLP hidden dimension $d_{\text{mlp}} = 128$ and $C = 11$ output classes. GELU is preferred over ReLU at this stage because its smooth, non-zero gradient near zero yields more stable gradient flow when MLP inputs are approximately centered [21]. The same classifier is applied independently to every frame, so the final output has shape $T \times C$.

Figure 5.1 illustrates the full three-stage pipeline.

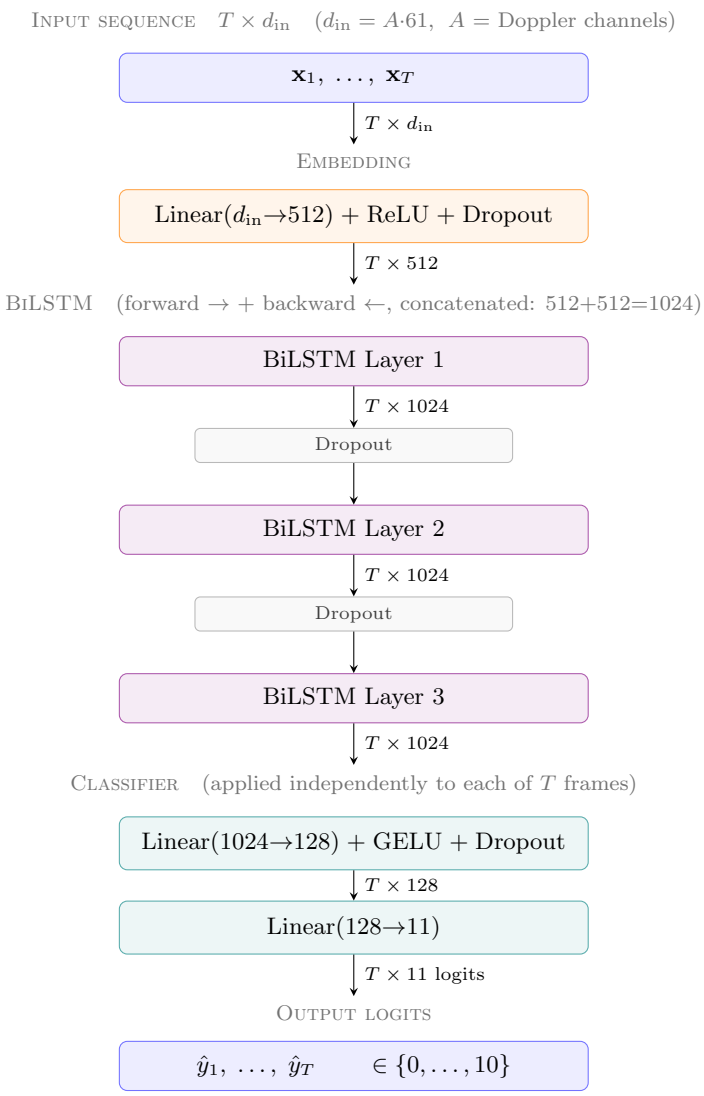


Figure 5.1: DopplerTAS data flow. Each input frame $\mathbf{x}_t \in \mathbb{R}^{d_{\text{in}}}$ ($d_{\text{in}} = A{\cdot}61$, where A is the number of Doppler channels, e.g. A=9 for the 3RX configuration) is first projected by a shared linear embedding. The full T-frame sequence then passes through three BiLSTM layers; each layer runs a forward and a backward LSTM pass over the sequence and concatenates their outputs, giving a 1024=2×512-dimensional representation per frame. Dropout is applied between consecutive layers. The two-layer classifier then maps each frame's 1024-d representation independently: first to 128 dimensions (GELU, dropout), then to 11-class logits. One prediction $\hat{y}_t$ is emitted for each of the T input frames.

5.1.3 Model Size

The total parameter count is approximately 23 M for the 3RX configuration ($d_{\text{in}} = 549$) and 22 M for 1RX ($d_{\text{in}} = 183$). The small differences arises because the embedding layer contributes relatively few parameters compared to the LSTM stack. One training epoch over the full 3RX training set takes approximately 420 s on a single NVIDIA RTX 3060 GPU.

5.1.4 Design Choices

No attention gating. A per-frame sigmoid gate over the BiLSTM output was evaluated as a design candidate. Such a gate introduces additional learned parameters that compete with the recurrent weights during optimization, while providing no additional effective temporal range beyond what the multi-layer BiLSTM already captures through its recurrent state. The empirical effect of adding this gate is reported in Section 6.2.2.

No layer normalization on recurrent outputs. Applying layer normalization [7] directly to the BiLSTM hidden states introduces a problematic inductive bias at initialization: normalizing recurrent states to unit variance before the weights have learned meaningful representations amplifies noise in the untrained states, which can cause the classifier to collapse to a near-uniform prediction. Empirically, this initialization instability outweighs the potential training-stability benefits. The effect on model accuracy is quantified in Section 6.2.2.

5.1.5 Training Protocol

During training, each recording is divided into overlapping windows of T consecutive Doppler frames using a sliding window with stride S. The window length T determines how much temporal context is available to the BiLSTM; the stride S controls the density of training samples per recording. At inference, predictions from overlapping windows are merged by taking the per-frame majority vote over all windows covering that frame. The effect of T on segmentation performance is examined in Section 6.2.2.

Features are z-score normalized per dimension using statistics computed from the training set. Training minimizes a weighted cross-entropy loss:

$$\mathcal{L} = -\frac{1}{T}\sum_{t=1}^{T} w_{y_t} \log \hat{p}_{t,\,y_t}, \qquad w_0 = 2.5, \quad w_{c>0} = 1.0, \tag{5.4}$$

where $\hat{p}_{t,c}$ is the softmax probability assigned to class c at frame t, y_t is the ground-truth label, and w_c is the per-class weight. The background class (class 0) is upweighted by a factor 2.5 because it comprises only 4–5% of frames; without upweighting the model tends to assign low loss to the majority activity classes and under-segment background transitions.

The optimizer is Adam [28] with initial learning rate $\eta = 5 \times 10^{-4}$ and weight decay $\lambda = 10^{-5}$. The learning rate is annealed with cosine scheduling [35] from η down to $\eta_{\min} = 10^{-5}$ over $T_{\max} = 200$ epochs. The checkpoint achieving the highest validation accuracy is retained for evaluation.

Chapter 6

Experimental Evaluation

This chapter reports all experimental results. Section 6.1 evaluates three baseline models — Wi-Monitor, WiFiTAD, and TW-FINCH — across four public datasets and the WiPos dataset using the benchmarking framework. Section 6.2 presents the DopplerTAS results on WiPos, including ablation studies on temporal context window and spatial diversity. Section 6.3 reports validation results on the activity recognition and localization dataset variants.

6.1 Benchmarking Framework Results

A key goal of this work is to provide a reusable benchmarking framework for WiFi-based temporal activity segmentation: a common data pipeline, evaluation protocol, and set of metrics that can be applied uniformly to any model and any dataset. To demonstrate the framework we evaluate three representative architectures from the literature — a supervised frame-level classifier (Wi-Monitor model [71]), a supervised temporal action detector (WiFiTAD [34]), and an unsupervised clustering method (TW-FINCH [45]) — on four public datasets and on the WiPos dataset (Chapter 3).

Beyond characterizing the baselines, the results on the WiPos dataset reveal a consistent accuracy drop compared to the public benchmarks, consistent with the position-induced covariate shift discussed in Section 5.1.1.

6.1.1 Models Under Evaluation

Wi-Monitor model. The Wi-Monitor model [71] is a supervised dense classifier; its architecture is described in Section 2.3.1. In brief, it processes fixed-size windows of raw CSI amplitude through a ResNet-2D encoder and a temporal convolutional network (TCN), producing per-frame class probabilities via a `Softmax` head. The paper describes a two-stage training procedure consisting of a base-training phase followed by a fine-tuning phase. The public code release does not fully specify the fine-tuning procedure, so our evaluation uses only the base-training stage. Training uses cross-entropy loss; the checkpoint with the highest validation accuracy is selected for testing.

WiFiTAD model. WiFiTAD [34] is a proposal-based temporal action detector adapted from the video domain; its underlying architecture is described in Section 2.3.1. Rather than producing a dense per-frame output, the model generates a set of temporal proposals, each described by a predicted start time, end time, and class score. Proposals are evaluated against ground-truth segments using intersection-over-union (IoU); detection quality is measured with mean average precision (mAP) at IoU thresholds {0.3, 0.5, 0.7}. No per-frame accuracy is reported because the output is not dense. The architecture requires a minimum input length of 2048 frames ($\approx 20.5\,\mathrm{s}$ at $100\,\mathrm{Hz}$).

TW-FINCH. TW-FINCH [45] is the unsupervised clustering baseline; its algorithm and CSI adaptation are described in Section 2.3.1. Because it produces unlabeled clusters rather than class predictions, the Hungarian algorithm is used to assign cluster indices to ground-truth class labels, yielding the best-case accuracy achievable by the discovered partition (an optimistic upper bound). Standard segmentation metrics are computed over the full dataset; there is no separate held-out test split.

Dataset and Evaluation Protocol

Five datasets are used for this benchmark:

Wi-Monitor A public indoor HAR dataset recorded with 802.11n commodity hardware. Activities are performed at fixed positions in front of a single router; the dataset contains 11 activity classes at a 100 Hz CSI sample rate with an average segment duration of approximately 1.78 s. Only test data was made publicly available; the original training set was not released. The test set contains duplicate samples; we removed duplicates and trained on the resulting unique samples, which yields a smaller effective training set than the original benchmark used. This reduction in training variety may partly explain lower scores compared to those reported in the original paper.

DeepSeg A public continuous activity segmentation dataset containing multiple subjects performing households activities in a residential environment. Activity classes are varied and the label distribution is notably imbalanced.

WiFiTAD The dataset accompanying the WiFiTAD temporal action detection model, designed explicitly for proposal-based evaluation with long, well-separated activity segments.

MM-Fi A large-scale multi-modal indoor dataset collecting synchronized WiFi CSI, mmWave radar, and RGB-D data for 27 activity classes. CSI is captured at 100 Hz; each activity instance lasts approximately 1.89 s. Each recordings consists of continuous repetitions of the same activity. The CSI amplitude data is stored in decibels ($20\log_{10}$ of the raw amplitude). Approximately 39% of segments contain $-\infty$ values in the amplitude. Inspection of the data shows the corruption is concentrated in specific recording sessions, suggesting a preprocessing error in the original MATLAB conversion pipeline. Removing the affected segments entirely is not a viable

fix: because the corruption is session-concentrated, dropping 39% of the data eliminates entire recording sessions rather than uniformly thinning the dataset. The result is a heavily session-biased and activity-unbalanced subset; we verified that training on such a reduced set produces similarly poor generalization. We therefore repair the affected entries by replacing each $-\infty$ value with the smallest finite value recorded for that subcarrier across the whole dataset (per-subcarrier column-minimum). This preserves the full session and class distribution, but introduces an artificial amplitude floor that does not reflect real channel conditions, which degrades the quality of the repaired data.

WiPos dataset Our new CSI dataset of 210 recordings (60 seconds each) collected with three receivers and motion capture for ground-truth frame-level labels. The subject performed 11 activity classes at 16 positions throughout the recordings space, not at a fixed position relative to the routers. CSI is recorded at 1000 Hz; the data is organized as three independent time series, each with per-frame shape $(3, 1, 114)$ (3 antennas, 1 stream, 114 subcarriers). For models that require a single combined input, the three streams are concatenated.

6.1.2 Wi-Monitor Model

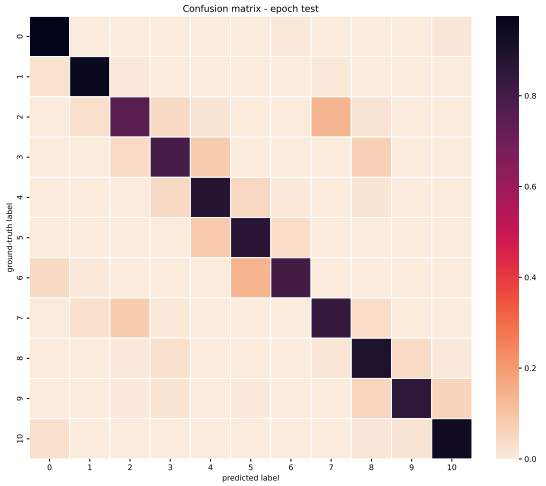
Table 6.1 summarizes the Wi-Monitor model test metrics across all five dataset conditions. Figure 6.1 shows the confusion matrix for the Wi-Monitor model trained and evaluated on the datasets.

Table 6.1: Wi-Monitor model test-set performance across datasets. mAP is at IoU = 0.7; F1 is Segmental F1 at IoU = 0.5; SegIoU = Symmetric Segment IoU; BndF1 = Boundary F1 (tol 500).

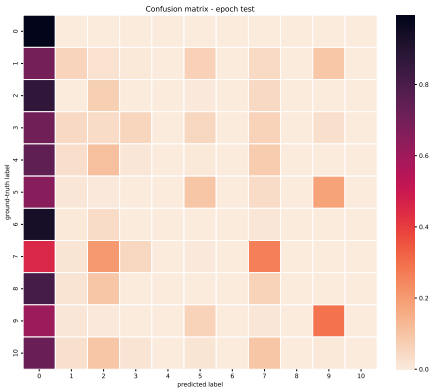
Dataset	Acc	mIoU	mAP @ 0.7	F1 @ 0.5	SegIoU	BndF1
Wi-Monitor	89.48%	77.41%	77.77%	77.59%	55.96%	53.14%
DeepSeg	66.76%	11.83%	2.35%	4.56%	39.80%	44.81%
WiFiTAD	29.87%	3.73%	0.00%	0.00%	35.52%	0.00%
WiPos	60.42%	48.40%	52.06%	25.56%	—	—
MM-Fi	4.19%	0.16%	0.00%	0.00%	—	—

Wi-Monitor dataset. On its native dataset the Wi-Monitor model achieves 89.48% frame accuracy and 77.41% mIoU, with Map@0.7 of 77.77%, F1@0.5 of 77.59%, Symmetric Segment IoU of 55.96%, and Boundary F1 of 53.14%. All six metrics are consistent: high accuracy is mirrored by strong mIoU, mAP, F1, Symmetric Segment IoU and Boundary F1, showing that the ResNet-2D + TCN architecture correctly identifies both activity boundaries as well as per-frame labels for the dataset it was designed for.

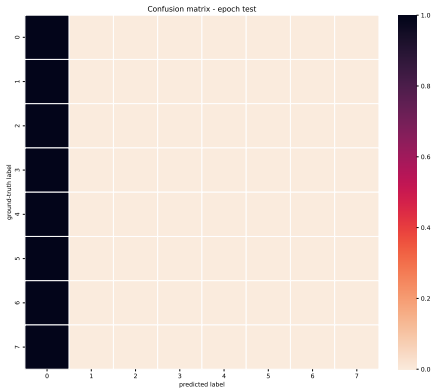
DeepSeg dataset. On DeepSeg, frame accuracy (66.76%) is much higher than mIoU (11.83%) and the near-zero mAP (2.35%) and F1 (4.56%). Figure 6.2 illustrates the failure mode: the ground-truth strip (top) contains vari-



(a) Wi-Monitor (native)



(b) DeepSeg



(c) WiFiTAD

Figure 6.1: Test-set confusion matrices for the Wi-Monitor model across three datasets. Rows are ground-truth labels, columns are predictions. The native Wi-Monitor result shows clean multi-class discrimination; the DeepSeg and WiFiTAD results exhibit near-total background collapse, with the entire prediction mass concentrated on the background column.

ous colored activity segments interspersed with background, while the prediction strip (bottom) is almost entirely blue (background) with only a few isolated non-background patches. This is background collapse: the model predicts background for nearly every frame regardless of what is actually happening, so it almost never detects the activity segments present in the ground truth. A predictor that outputs background at every frame achieves roughly 67% frame accuracy on DeepSeg — matching the 67% background proportion of the dataset — while getting near-zero IoU, mAP, and F1, since it detects no segments at all. The Wi-Monitor model was trained on its native dataset where background is a minority class, so no pressure exists during training to suppress background predictions when transferred to the background-dominated structure of DeepSeg.

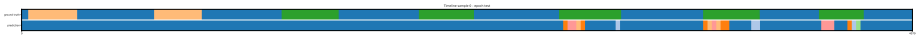


Figure 6.2: Ground-truth (top) and predicted (bottom) activity timeline for a representative DeepSeg test recording, Wi-Monitor model. Blue is class 0 (background), other colors are activity classes. The ground-truth strip contains substantial colored activity segments; the prediction strip is almost entirely blue, with only a few isolated non-background patches — confirming background collapse on this dataset. The white horizontal bar separates the two rows.

WiFiTAD Dataset On the WiFiTAD dataset, the Wi-Monitor model achieves only 29.87% accuracy with zero mAP and F1, despite being trained on the WiFiTAD training split. The WiFiTAD dataset is structured differently from Wi-Monitor: each recording is a long window containing one activity surrounded by a substantial period of background with no labeled activity, while the Wi-Monitor architecture is designed for short, densely annotated segment sequences where activity frames are the majority class. Even with in-domain training, the frame-level classifier cannot reliably suppress the dominant background class, so predictions collapse to background and activities are rarely detected, yielding near-zero mAP and F1 alongside a frame accuracy only slightly above chance.

WiPos Dataset The Wi-Monitor model achieves 60.42% frame accuracy on the WiPos dataset, with mIoU of 48.40% and mAP@0.7 of 52.06%. Unlike the DeepSeg result, all four metrics are consistent, which shows the model is making real per-class predictions rather than collapsing to a single dominant class.

The 60.42% accuracy is lower than the 89.48% achieved on the Wi-Monitor dataset, despite both datasets having the same number of classes. We hypothesize that this gap is mainly due to the positional variation in the WiPos dataset: in the Wi-Monitor recordings, activities are always performed at the same location in front of a single router, so each class produces a relatively stable CSI amplitude pattern across sessions. In the WiPos dataset, the subject performs the same activities at 16 different positions, so the CSI amplitude pattern — which depends on the full multipath channel and therefore on position — varies substantially for the same activity.

MM-Fi Dataset The Wi-Monitor model achieves only 4.19% frame accuracy on MM-Fi, barely above the 3.7% random-chance baseline for 27 classes.

As described above, the MM-Fi amplitude data contains $-\infty$ values for null subcarriers. Even after we replace these with column-minimum values, the repaired samples are not equivalent to clean measurements: the replaced values introduce an artificial amplitude floor that does not reflect real channel conditions. Training accuracy reaches approximately 53% while validation accuracy plateaus at 9.5%, indicating the model memorizes the training set but does not generalize. The 27-class label space combined with the corrupted input signal makes MM-Fi a difficult dataset for this model.

6.1.3 WiFiTAD Model

Table 6.2 reports mAP at three IoU thresholds and Segmental F1 for the WiFiTAD model. No result is reported for MM-Fi; see below.

Table 6.2: WiFiTAD model test-set results across datasets. Segmental F1 is at IoU = 0.5. MM-Fi is excluded due to a tensor-shape incompatibility caused by the minimum 2048-frame input requirement of the WiFiTAD architecture and the ≈189-frame average MM-Fi activity duration.

Dataset	mAP @ 0.3	mAP @ 0.5	mAP @ 0.7	F1 @ 0.5
DeepSeg	87.68%	84.42%	75.38%	1.59%
Wi-Monitor	**99.72%**	**98.14%**	**92.08%**	**2.73%**
WiFiTAD	75.88%	69.37%	52.69%	1.48%
WiPos	71.25%	54.41%	35.66%	2.32%

Wi-Monitor and DeepSeg datasets. WiFiTAD achieves exceptional mAP on Wi-Monitor (mAP@0.7 = 92.08%) and strong mAP on DeepSeg (75.38%), demonstrating that the multi-scale temporal proposal architecture localizes activity boundaries precisely when the CSI signal has sufficient temporal variation.

The Segmental F1 remains near zero on both datasets (2.73% and 1.59%), which seems contradictory given the high mAP. The discrepancy arises from how each metric counts a correct prediction: mAP rewards any high-overlap proposal, so even a few well-placed proposals yield a high score. Segmental F1 requires a one-to-one match for every ground-truth segment; WiFiTAD produces a small number of large proposals, while these datasets contain many short segments, so most ground-truth segments go unmatched. Whether this is a failure depends on the goal: for localizing when activities occur in a long recording, the high mAP is the relevant result; for labeling every individual repetition, the near-zero F1 shows the model falls short.

WiFiTAD dataset. On its own test set WiFiTAD achieves mAP@0.7 = 52.69% and mAP@0.5 = 69.37%. This is lower than on Wi-Monitor and DeepSeg, which is possibly explained by the WiFiTAD dataset's smaller size and more varied activity lengths, both of which can increase the variability of localization scores.

WiPos dataset. On the WiPos dataset WiFiTAD achieves mAP@0.3 = 71.25% but drops to mAP@0.7 = 35.66%. The gap suggests the model can roughly locate an activity within a recording but cannot reliably align its predicted start and end times with the precise frame-level labels we provide. WiFiTAD generates a small number of coarse proposals per recording, each covering the rough time span of one activity, while the annotations of WiPos mark boundaries at the individual frame level. The structural mismatch between few coarse proposals and many precise frame-level boundaries likely explains the drop in mAP as the IoU threshold increases.

MM-Fi dataset. We exclude MM-Fi from the WiFiTAD benchmark. WiFiTAD requires a minimum input of 2048 frames (≈ 20.5 s at 100 Hz) for its temporal feature pyramid. The average MM-Fi activity instance lasts only 189 frames, well below this minimum. Padding or truncating samples to 2048 frames would change the temporal statistics of the input fundamentally, making the comparison meaningless.

TW-FINCH Model

Table 6.3 summarizes TW-FINCH results. Because TW-FINCH is unsupervised it requires a different evaluation strategy from the supervised models. There is no training phase; the algorithm runs on the complete dataset and produces unlabeled clusters. It then applies the Hungarian algorithm [29] to assign each cluster index to the ground-truth class label that maximizes overall accuracy. This optimal label assignment represents the best-case scenario; to put the resulting numbers in context, we also report a random-prediction baseline in Table 6.3: applying Hungarian matching to completely random cluster assignments on the WiPos dataset yields 50.38% accuracy, 33.84% mIoU, 12.63% mAP@0.7, and 39.04% F@0.5. There is no held-out test split; all metrics are over the complete dataset.

Table 6.3: TW-Finch unsupervised clustering results after Hungarian label assignment. All metrics are over the complete dataset; there is no held-out test split because TW-Finch is unsupervised. SegIoU = Symmetric Segment IoU; BndF1 = Boundary F1 (tol 500); both are only reported for datasets where ground-truth segments are fully defined (Wi-Monitor and DeepSeg). We exclude MM-Fi because TW-Finch fails to produce the required number of clusters on any MM-Fi recording (see Section 6.1.3). The last row shows a random-prediction baseline (random cluster assignments with optimal Hungarian label assignment applied) on the WiPos dataset.

Dataset	**Acc**	**mIoU**	**mAP@0.7**	**F1@0.5**	**SegIoU**	**BndF1**
Wi-Monitor	59.85%	41.42%	17.79%	32.53%	54.44%	58.88%
DeepSeg	31.54%	17.97%	17.45%	38.71%	60.96%	78.31%
WiFiTAD	47.62%	30.41%	5.28%	3.60%	—	—
WiPos	52.29%	35.43%	16.38%	16.65%	—	—
Random	*50.38%*	*33.84%*	*12.63%*	*39.04%*	—	—

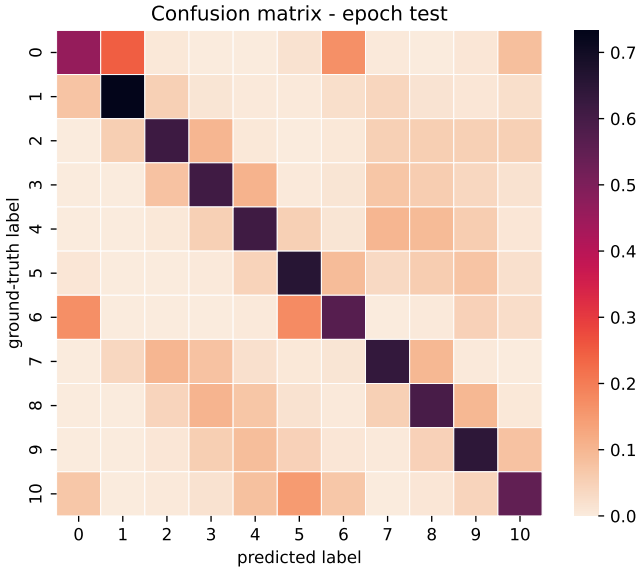


Figure 6.3: Confusion matrix for TW-FINCH on the Wi-Monitor dataset after Hungarian label assignment. Rows are ground-truth classes, columns are the assigned cluster labels. Off-diagonal mass indicates classes that the pairwise affinity does not fully separate.

Wi-Monitor dataset. TW-FINCH achieves 59.85% frame accuracy on Wi-Monitor without any supervision, with a Segmental F1 of 32.53%, Symmetric Segment IoU of 54.44%, and Boundary F1 of 58.88%. The time-weighted affinity discovers temporal clusters that align well enough with the ground-truth activity classes for the Hungarian assignment to produce a meaningful label mapping.

DeepSeg and WiFiTAD datasets. Performance on WiFiTAD (47.62%) and DeepSeg (31.54%) is lower. The Segmental F1 on Wi-Monitor (32.53%) is much higher than on WiFiTAD (3.60%), which can be explained by dataset structure: TW-FINCH produces a dense, gap-free partition of each recording. This fits naturally with the dense short-activity structure of Wi-Monitor, but fits poorly with the WiFiTAD structure where activities are surrounded by long background periods. The mAP@0.7 scores across all datasets are low (5–18%), reflecting that while TW-FINCH identifies the correct activity for many frames, the cluster boundaries rarely align precisely with ground-truth segment boundaries at the 70% IoU threshold.

WiPos dataset. TW-FINCH achieves 52.29% accuracy and 35.43% mIoU on the WiPos dataset, approaching the supervised Wi-Monitor model on the same data (60.42% acc, 48.40% mIoU). That an unsupervised algorithm ties with a trained classifier indicates that the 11 activities in the recordings produce sufficiently distinct temporal CSI signatures for the pairwise affinity to separate them without labels.

MM-Fi dataset. We exclude MM-Fi from the TW-FINCH results. TW-FINCH's hierarchical partition cannot produce the 27 required clusters on any MM-Fi recording: the algorithm reaches maximum granularity with at most 25 groups per recording. The likely cause is that each MM-Fi activity instance lasts only $\approx$1.89 s on average, which is probably too short to accumulate enough temporal variation to form 27 distinct groups. The clusters that are produced have near-zero detection metrics (mAP@0.7 = 0.00%, Segmental F1 = 0.00%, mIoU = 7.21%), confirming the clustering is not informative on this dataset.

6.1.4 Cross-Model Comparison and Discussion

Table 6.4 provides a unified side-by-side comparison of the three baselines on the datasets for which all models produced valid results.

Table 6.4: Cross-model comparison of the three baselines across six metrics. WM = Wi-Monitor model, WT = WiFiTAD model, TWF = TW-Finch. "—" means the metric is not applicable to the model or was not recomputed for that condition. SegIoU = Symmetric Segment IoU; BndF1 = Boundary F1 (tolerance 500 frames). WiPos SegIoU/BndF1 omitted (dataset snapshot used for original run is no longer available). All values are percentages.

Dataset	Model	Acc	mIoU	mAP@0.7	F1@0.5	SegIoU	BndF1
Wi-Monitor	WM	89.48	77.41	77.77	77.59	55.96	53.14
	WT	—	—	92.08	2.73	—	—
	TWF	59.85	41.42	17.79	32.53	54.44	58.88
DeepSeg	WM	66.76	11.83	2.35	4.56	39.80	44.81
	WT	—	—	75.38	1.59	—	—
	TWF	31.54	17.97	17.45	38.71	60.96	78.31
WiPos	WM	60.42	48.40	52.06	25.56	—	—
	WT	—	—	35.66	2.32	—	—
	TWF	52.29	35.43	16.38	16.65	—	—

mAP and frame accuracy measure different qualities. WiFiTAD consistently achieves the highest mAP on datasets with sufficiently long sequences (Wi-Monitor: 92%, DeepSeg 75%) but reports near-zero Segmental F1. The Wi-Monitor model achieves the highest frame accuracy and F1 on its native dataset (88% and 70%) while WiFiTAD has no frame accuracy at all. These are not contradictory: the two architectures optimise for structurally different output spaces. mAP evaluates the ranked precision of temporal proposals, which WiFiTAD excels at; frame accuracy and F1 evaluate dense per-frame labeling, which the Wi-Monitor model is designed for. Choosing which metric is most meaningful depends on the target application.

Accuracy degradation on the WiPos dataset and its implications All baselines achieve lower performance on WiPos than on Wi-Monitor despite both sharing 11 classes: the Wi-Monitor model drops from 89.48% to 60.42%, WiFiTAD's mAP@0.7 falls from 92.08% to 35.66%, and TW-FINCH from 59.85% to 52.29%. We primarily attribute this to the positional variation in WiPos: while

Wi-Monitor activities are always performed at the same location, WiPos activities are at 16 different positions, causing the CSI amplitude pattern to vary with position (see Section 2.1), consistent with the covariate shift discussed at the start of this section. An additional contributing factor is that the baselines were evaluated with their original published hyperparameters, optimized for their respective native datasets; hyperparameter tuning specific to WiPos could partially close the gap, though it would not eliminate the underlying position-sensitivity of amplitude-based features.

Figure 6.4 shows four out of six metrics (Acc, mIoU, mAP@0.7, F1@0.5) across all four datasets and all three models in a single overview.

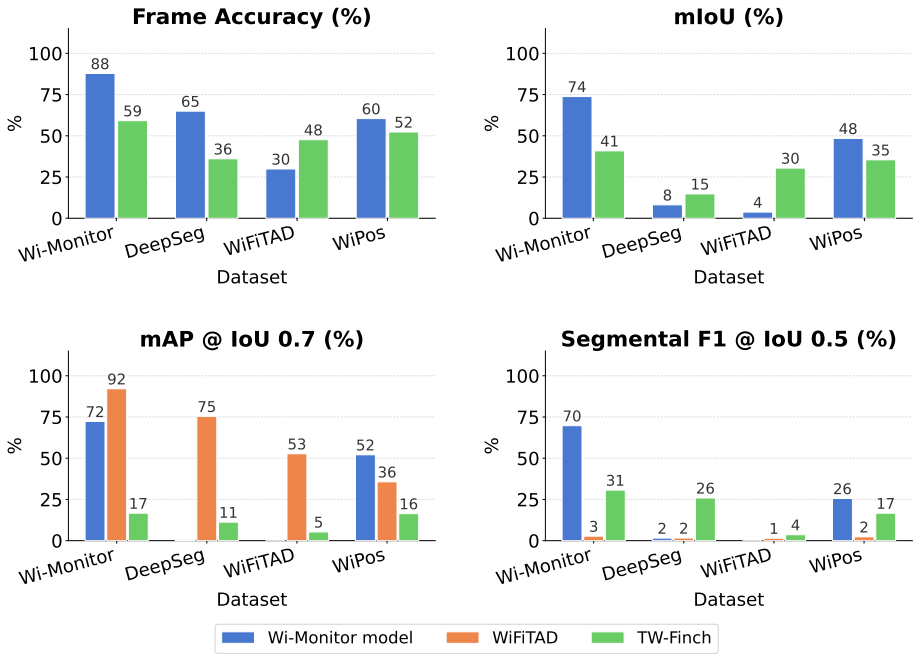


Figure 6.4: Benchmarking framework results for all three baselines across all four datasets. Each panel shows one metric; bars are grouped by dataset and colored by model. WiFiTAD reports no dense Acc or mIoU and is absent from those panels. MM-Fi is excluded (near-zero scores due to corrupted input; see Section 6.1.2).

6.2 DopplerTAS Model Results

All experiments in this section use the DopplerTAS model described in Section 5.1. The model is evaluated on the 3RX Doppler dataset by default, with one benchmark on the 1RX Doppler dataset for comparison.

6.2.1 Evaluation Metrics

The held-out test split comprises 509 sliding windows (15% of 3,391 total), drawn at the window level from across all recordings using a fixed seed, so test windows

span the full diversity of recording sessions. Six complementary metrics are reported:

Frame accuracy Fraction of frames classified correctly.

mIoU Mean intersection-over-union across all 11 classes.

mAP @ IoU 0.7 Mean average precision at a 70% IoU threshold, computed from per-segment confidence scores derived from the dense logit output.

Segmental F1 @ IoU 0.5 F1 score for one-to-one segment matching at a 50% IoU threshold.

Segment IoU (symmetric) Jaccard index at the segment level, averaged symmetrically over ground-truth and predicted segments.

Boundary F1 (tol = 5) F1 score for predicted vs. true segment boundaries with a tolerance of 5 frames.

6.2.2 Effect of Temporal Context Window

The dominant design variable in the DopplerTAS pipeline is the training window size T. Table 6.5 reports the effect of varying T with all other hyperparameters held constant.

Table 6.5: DopplerTAS test-set performance as a function of training window size T on the 3RX Doppler temporal dataset. All runs use z-score normalization, weighted cross-entropy, Adam ($\eta = 5 \times 10^{-4}$, $\lambda = 10^{-5}$), and cosine annealing over 200 epochs; stride $S = 300$ throughout. **Ep**: epoch at which the best-validation-accuracy checkpoint was saved.

Window T	Test Acc	Best Val	Ep	mIoU
500	76.18%	76.81%	138	59.71%
750	82.21%	83.36%	143	65.78%
1000	86.26%	85.38%	145	71.14%
1500	**96.70%**	**97.08%**	**188**	**90.43%**

Each 250-frame increase in context delivers approximately 4–10 percentage points of test accuracy. We attribute this to two complementary effects. First, larger windows allow the BiLSTM to observe more complete action instances, reducing truncated-action artifacts near window edges. Second, boundary frames benefit from proportionally more surrounding context on both sides, which directly helps the backward LSTM pass resolve ambiguous onset/offset timing.

The full metrics for the best configuration ($T = 1500$) are reported in Table 6.6.

6.2.3 Spatial Diversity: 3RX vs. 1RX

To quantify the contribution of multi-receiver spatial diversity, we train an identical model on the 1RX dataset (middle receiver only) with all other settings unchanged. The experiment is a controlled ablation: any accuracy difference is attributable solely to the reduction from three receivers to one.

Table 6.6: Full test-set metrics for DopplerTAS on the 3RX dataset ($T = 1500$, $S = 300$, 200 epochs).

Metric	Value
Frame accuracy	96.70%
mIoU	90.43%
mAP @ IoU 0.7	90.32%
Segmental F1 @ IoU 0.5	85.08%
Segment IoU (symmetric)	97.25%
Boundary F1 (tol = 5)	87.47%
Best-checkpoint epoch	188 / 200

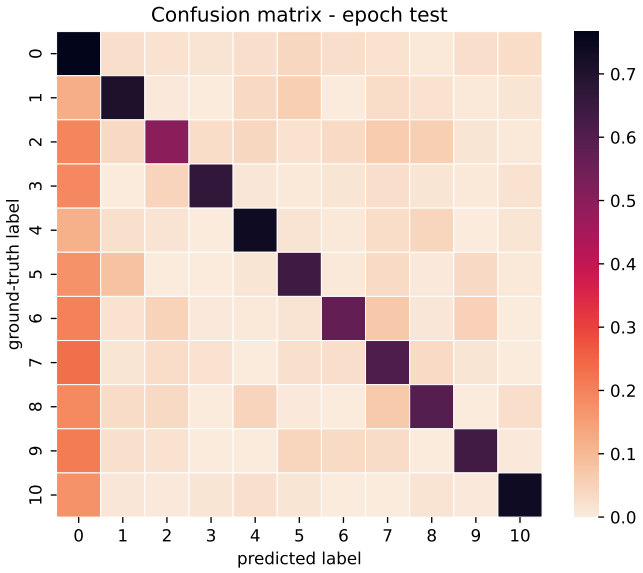


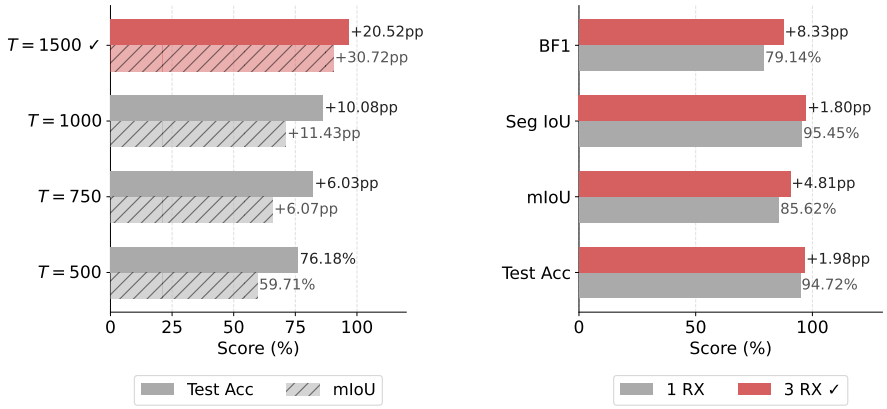
Figure 6.5: Test-set confusion matrix for DopplerTAS 3RX ($T = 1500$). Rows are ground-truth labels, columns are predictions; values are proportional to the number of frames.

Table 6.7: 3RX vs. 1RX comparison on the temporal segmentation task (DopplerTAS, 3RX, $T = 1500$, $S = 300$).

Setup	Test Acc	mIoU	Seg. IoU	BF1
3RX ($T = 1500$)	**96.70%**	**90.43%**	**97.25%**	**87.47%**
1RX	94.72%	85.62%	95.45%	79.14%

The results reveal a surprisingly small performance gap. Restricting to a single receiver reduces frame accuracy by only 1.98 percentage points (96.70% → 94.72%) and mIoU by 4.81 pp (90.43 → 85.62%). This minimal gap demonstrates that the critical mechanism for position invariance in DopplerTAS is the antenna-conjugation operation rather than the number of receivers: even a single router's three antenna channels produce a largely position-independent Doppler velocity estimate. The metrics remain high for the 1RX configuration, with Segment IoU at 95.45% and Boundary F1 at 79.14%, confirming that a single receiver is sufficient for reliable temporal segmentation.

Figure 6.6 summarizes the two ablation dimensions side by side. Panel (a) shows the monotonic gain in Test Acc and mIoU as the training window grows from $T = 500$ to $T = 1500$; panel (b) shows the per-metric improvement when the spatial diversity is extended from one receiver to three.



(a) Effect of training window size T on Test Accuracy and mIoU (all other settings fixed, 3 RX). Annotations show the gain relative to the $T = 500$ baseline; $T = 1500$ is the chosen configuration (✓).

(b) Impact of spatial diversity: 1 RX versus 3 RX across all four test-set metrics ($T = 1500$). Annotations show the absolute gain of the 3 RX configuration.

Figure 6.6: DopplerTAS ablation study.

6.2.4 Discussion

The 96.70% frame accuracy achieved by DopplerTAS (3RX, $T = 1500$) is notable for two reasons.

First, the improvements were obtained without any architectural changes; only the training window size was increased. This indicates that the BiLSTM model capacity was never the bottleneck: the model was always capable of discriminating the 11 activity classes given sufficient temporal context.

Second, the Doppler representation is inherently position-invariant. Because Doppler frequency shifts depend on radial velocity rather than absolute position in the room, the model generalizes across recording sessions without requiring any position-aware calibration. This contrasts with models trained on raw CSI

amplitude, where the spatial distribution of the subject relative to the antenna array varies between sessions and introduces covariate shift.

The 3RX vs. 1RX ablation reveals a surprisingly small performance gap between receiver configurations. All metrics remain high for the 1RX configuration, with frame accuracy rising by only 1.98 pp. This suggests that a single-receiver deployment is a viable practical option: the dominant driver of position invariance is the antenna-conjugation operation, not the number of receivers. The 94.72% single-receiver accuracy is only slightly below the three-receiver result, making single-receiver deployment attractive for scenarios where hardware cost or installation simplicity is a priority.

Despite the high accuracy, the training dynamics also reveal a discernible degree of overfitting. By the final epoch (200/200) training loss had decreased to 0.0036 — a near-zero plateau — while validation loss remained at 0.31, an approximately 85-fold gap. Training accuracy reached 99.85% while the best validation accuracy was 97.08%, achieved at epoch 188; the remaining 12 epochs produced no further improvement on the validation set. In this experiment best-checkpoint selection already mitigates the effect: the reported test result of 96.70% is evaluated from the epoch-188 checkpoint rather than the final model. Nevertheless, the loss divergence indicates that training resources are being spent on further memorization long after generalization has saturated. Future work could replace the fixed 200-epoch schedule with early stopping tied to validation loss, or explore stronger regularization — increased dropout in the recurrent layers and additional weight decay — to slow loss divergence and potentially reach the validation optimum at a lower loss level.

Figure 6.7 places DopplerTAS in the context of the three baselines evaluated on WiPos. WiFiTAD produces no dense per-frame output and therefore cannot report frame accuracy or mIoU; those two cells are marked "n/a". All other metric–model combinations are valid. On the two metrics shared by the dense classifiers — frame accuracy and mIoU — DopplerTAS (96.7% / 90.4%) improves substantially over the best baseline, the Wi-Monitor model (60.4% 48.4%), confirming the advantage of the position-invariant Doppler representation. On the proposal-level metrics, DopplerTAS achieves mAP@0.7 = 90.3% and Segmental F1 = 85.1%, far exceeding every baseline on both measures.

6.2.5 DopplerTAS on Wi-Monitor Dataset

To assess whether the Doppler feature pipeline generalizes beyond the WiPos dataset, we trained DopplerTAS from scratch on the Wi-Monitor dataset, where all experimental settings, such as architecture, optimizer, and loss function, were kept identical to the WiPos 3RX run.

At epoch 117 a validation accuracy of 85.34% had been reached. Table 6.8 reports the full test-set metrics.

DopplerTAS achieves 82.15% frame accuracy and 65.22% mIoU on the Wi-Monitor dataset. Compared with the native Wi-Monitor model (Section 6.1.2), which achieves 89.49% accuracy on the same data, DopplerTAS trails by 7.33 pp. One factor plausibly accounts for this gap: DopplerTAS's architecture and key hyperparameters were tuned on WiPos data and applied to Wi-Monitor without re-optimization; the native Wi-Monitor model is purpose-built and tuned for that distribution. Despite this gap, the result demonstrates that the DopplerTAS pipeline is hardware-agnostic in the sense that it can be retrained

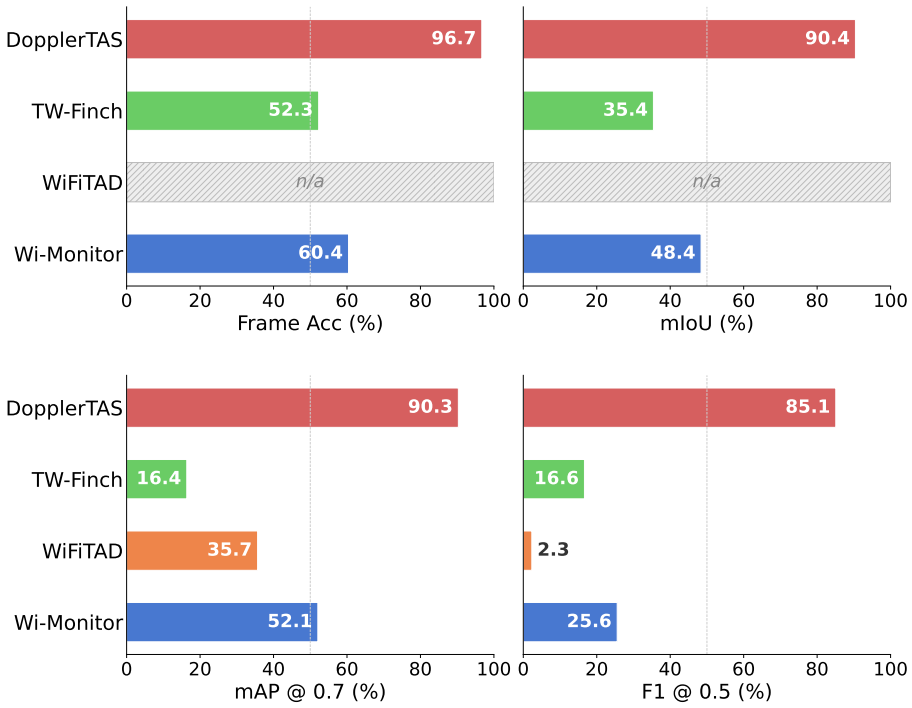


Figure 6.7: WiPos test-set results for all four models. Each panel shows one metric; the hatched "n/a" cells for WiFiTAD in the Acc and mIoU panels indicate that WiFiTAD produces no dense per-frame output and therefore cannot report those metrics. WM = Wi-Monitor, WT = WiFiTAD, TWF = TW-Finch, DT = DopplerTAS (3RX, $T = 1500$).

on different data and achieve competitive performance without any architectural changes.

6.3 Validation on Joint Activity Recognition and Localization

To validate the quality and utility of our motion capture labels on a task beyond temporal segmentation, we evaluate the derivative activity recognition and localization dataset (described in Section 3.7) using the ARIL model architecture [54]. This experiment serves two purposes: (1) demonstrating that the position labels extracted from motion capture data are sufficiently accurate to train a localization model, and (2) exploring the behavior of WiFi CSI-based recognition under a within-session evaluation protocol.

Table 6.8: DopplerTAS test-set metrics on the Wi-Monitor dataset ($T = 1500$; early-stopped at epoch 117).

Metric	Value
Frame Accuracy	82.15%
mIoU	65.22%
mAP (@0.7)	76.64%
Segmental F1 (@0.5)	73.61%
Seg. IoU Symmetric	85.02%
Boundary F1 (tol = 50)	76.04%

6.3.1 Experimental Setup

Model Architecture

We adopt the 1D ResNet architecture from the original ARIL work [54], with layer configuration [1,1,1,1] using BasicBlock. The model has two output heads:

- **Activity head**: 8-class softmax classifier
- **Position head**: 16-class softmax classifier

The input is CSI amplitude of a single receiver antenna, yielding a tensor of shape $(56, 700)$ — 56 subcarriers $\times$ 700 time-steps. Antenna 0 of the middle receiver is selected; using a single antenna (rather than concatenating all four) keeps the input dimensionality consistent with the original ARIL formulation and reduces model complexity. Both heads are trained jointly with cross-entropy loss.

Train/Test Split

We use a **random window-level split**: windows are shuffled and assigned to train (80%), validation (10%), and test (10%) sets independently of the recording session they originate from. This yields:

- 4,656 windows in the training set
- 518 windows in the validation set
- 575 windows in the test set

Because windows from the same recording session can appear in both training and test sets, this protocol does not evaluate cross-session generalisation. The reported accuracy therefore reflects within-session performance, which is an upper bound on the true generalisation ability of the model. A session-held-out evaluation is discussed in Section 6.3.3.

Normalization

Z-score normalisation is applied across all training examples per channel and propagated to the validation and test folds.

Training Configuration

- Optimizer: Adam, learning rate 0.005
- LR schedule: MultiStepLR, decay by 0.5 every 10 epochs
- Batch size: 128
- Epochs: 200

6.3.2 Results

Table 6.9 shows accuracy on both tasks at key epochs; the best validation activity accuracy occurred at epoch 163.

Table 6.9: ARIL Model Accuracy — Train vs. Validation (Random Window-Level Split).

Epoch	Activity Train	Activity Val	Position Train	Position Val
0	14.3%	14.9%	29.6%	30.9%
10	31.8%	29.0%	68.2%	67.8%
30	65.5%	55.0%	88.7%	81.3%
80	91.0%	62.2%	97.3%	87.5%
163	**91.6%**	**62.7%**	**97.8%**	**86.7%**
199	92.2%	60.6%	97.4%	86.1%

Table 6.10: Final Test Performance (Epoch 200).

Task	Test Accuracy	Chance Level
Activity Recognition (8 classes)	57.0%	12.5%
Indoor Localization (16 classes)	87.7%	6.25%

6.3.3 Analysis

Within-session performance. The model reaches approximately 92% training accuracy and 63% validation accuracy on activity recognition, with a smaller gap on localization (97% vs. 87%). The narrower train–val gap compared to a session-held-out evaluation is expected: windows from the same session share channel characteristics, so part of what the model has memorized transfers trivially to held-out windows from the same session.

Task difficulty disparity. Localization (87.7% test) substantially outperforms activity recognition (57.0% test), despite having twice as many classes (16 vs. 8). This is consistent with the hypothesis that spatial position creates stable, position-dependent multipath signatures that the model can exploit, while activity variations within a position are subtler and more session-dependent.

Label quality validation. Both tasks comfortably exceed random chance (12.5% and 6.25% respectively). The 87.7% localization accuracy under the random split, and the strong learning curve visible in Table 6.9, indicate that the motion capture-derived position labels are spatially accurate and temporally aligned with the CSI data.

Caveat: within-session data leakage. Because the random split does not respect session boundaries, the test accuracy is not a reliable measure of generalisation to new recording sessions. When the same protocol is evaluated with a session-held-out split (i.e. no session overlap between train and test), activity recognition degrades substantially, which is consistent with the well-known session-dependent nature of WiFi CSI. The localization task degrades less severely because coarse position signatures are more stable. Addressing cross-session generalisation requires domain-adaptation techniques or considerably more diverse training data.

6.3.4 Comparison to Temporal Segmentation

ARIL is the only model in this chapter that is not a temporal segmentation method: it was included as a dataset validation experiment — specifically to validate the motion-capture-derived position labels — rather than to benchmark segmentation performance. Unlike every other model evaluated here, which must infer activity boundaries from a continuous CSI stream, ARIL receives pre-segmented windows with known boundaries. This task formulation difference is why it is compared separately rather than alongside the temporal segmentation results; the within-session activity recognition accuracy of 57% is nevertheless instructive: even when windows from the same session appear in both folds, activity classification is harder than localization, highlighting the richness of spatial information encoded in CSI and the relative subtlety of activity-induced channel variation.

6.3.5 Summary

The primary finding of the ARIL experiment is that the motion-capture-derived position labels are high quality: the 87.7% localization accuracy confirms that the labels are spatially accurate and temporally aligned with the CSI data.

Activity recognition reaches only 57% under the same within-session split. Inspection of the per-class confusion matrix reveals that the errors are not evenly distributed: two activity clusters account for the majority of misclassifications. The first is {squat, jumping jack, jump}, three activities that all involve rapid whole-body vertical displacement and are frequently predicted as one another. The second is {boxing, stir a pot, raising left arm}, three activities that share a repetitive upper-arm oscillation pattern. This is a fundamentally harder inter-class discrimination problem than the original ARIL benchmark, which distinguishes six geometrically distinct hand gestures (up, down, left, right, circle, cross).

Chapter 7

Discussion

This chapter reflects on the key findings from the dataset creation, benchmarking evaluation, and DopplerTAS model, discusses the limitations of this work, and contextualises within the broader WiFi sensing landscape.

7.1 Key Findings

7.1.1 Dataset Quality and Annotation Precision

The most fundamental design choice in WiPos is the use of motion capture as the ground-truth source for temporal labels. Unlike the video-based labeling used in prior datasets (see Section 2.3.4), boundaries in WiPos are derived accurately from body-marker motion capture data sampled at 960 Hz, using a specifically designed annotation tool, reducing annotation error to the millisecond scale. The consequence is directly observable in the ARIL localization experiment: the localization achieves 87.7% accuracy across 16 spatial classes, indicating that the position labels are precise enough for the model to learn clean spatial signatures. Labels that were temporally or spatially imprecise would produce blurred class boundaries and decrease this accuracy.

The continuous recording protocol — randomized activity order, variable durations of 3–10 s, no pauses between activities — also proved consequential. Models that rely on amplitude thresholds to detect activity onset (implicitly assuming brief silence between activities) fail on WiPos because no such silence is present. This design choice faithfully reflects real-world deployment scenarios and separates models that have genuinely learned temporal dynamics from those that exploit inter-activity gaps.

7.1.2 Model Performance Characteristics

The most striking result from the benchmarking evaluation is the consistent degradation all three baseline models exhibit when moving from their native datasets to WiPos. The Wi-Monitor model drops from 89.48% to 60.42% frame accuracy; WiFiTAD's mAP@0.7 falls from 92.08% to 35.66%; TW-FINCH drops from 59.85% to 52.29%. These drops across three different architectures — supervised dense classification, proposal-based detection, and parameter-free clustering — rules out model-specific explanations and points instead to a shared

input characteristic: all three models operate on raw CSI amplitude, which encodes the multipath channel jointly as a function of motion and position. When subjects move freely across 16 positions, the same activity at different positions produces different amplitude patterns, constituting a covariate shift that amplitude-based models cannot overcome without positional supervision.

DopplerTAS directly addresses this by operating on Doppler features derived from the time-differential phase. Because Doppler shifts measure radial velocity rather than path length, they are largely position-independent, and the model achieves 96.70% frame accuracy on WiPos — 30 percentage points above the best amplitude-based baseline. This demonstrates that representation choice, not model capacity, was the constraint: the three-layer BiLSTM is architecturally simpler than the ResNet-2D + TCN of the Wi-Monitor model, yet substantially outperforms it on this dataset.

Two ablations clarify which design dimensions matter most (Tables 6.5 and 6.7). The window ablation shows that temporal context is the dominant design variable: each 250-frame increase reliably improves accuracy, with the full $T = 500 \rightarrow T = 1500$ range spanning a 20.5 pp gain. The receiver ablation reveals the opposite: restricting to a single receiver costs only 1.98 pp, confirming that the dominant driver of position invariance is the antenna-conjugation operation rather than spatial receiver diversity.

7.1.3 Insights Enabled by the Benchmarking Framework

Several findings were only possible because the evaluation was conducted through a standardized framework:

- **Fair comparison reveals shared root cause.** Because all three baselines were trained and evaluated with identical preprocessing pipelines, data splits, and metrics, the consistent WiPos accuracy drop cannot be attributed to different normalization choices or evaluation protocols. The framework's standardized conditions transform an observation into a controlled finding: the drop is caused by position-induced covariate shift, not experimental variation.

- **Different metrics expose different failure modes.** WiFiTAD achieves mAP@0.7 = 92% on the Wi-Monitor dataset while its Segmental F1 is near zero; the Wi-Monitor model achieves 88% frame accuracy but produces no proposal-level detections. Without a framework computing all four metrics consistently, these complementary failure modes remain hidden behind single-metric comparisons.

- **Preprocessing isolation:** The framework's configurable preprocessing pipeline makes it possible to test different normalization strategies without changing any other part of the evaluation protocol, isolating their individual contribution.

- **Reproducibility.** Fixed random seeds in the data-splitting and training loops mean that all results in thesis thesis can be reproduced exactly from the same codebase and configuration files.

7.1.4 Implications for Future Research

Dataset design. Based on the WiPos collection experience and the experimental results, we recommend three design priorities for future CSI segmentation datasets: (1) motion-capture or equivalent high-rate ground truth for annotation rather than video labeling, (2) continuous activity sequences without artificial static transitions, and (3) spatial diversity through either free subject movement or systematic position variation, to prevent models from learning position-specific channel signatures.

Evaluation practices. The divergence between mAP and frame accuracy in this work shows that reporting a single metric is insufficient for temporal segmentation. Proposal-based metrics (mAP) capture localization quality; dense-prediction metrics (frame accuracy, mIoU) capture classification completeness; boundary-level metrics (Boundary F1) capture onset and offset precision. A complete evaluation should include at least one metric from each category.

Model design. The success of DopplerTAS suggests that position-invariant feature representations should be a design concern for any model intended to operate in free-movement environments. The strong performance dependence on training window size further suggests that future architectures should prioritize long-range temporal modeling, whether through larger windows, hierarchical encoders, or attention mechanisms.

7.2 Limitations

While this work provides foundational infrastructure for WiFi-based activity segmentation research, several limitations should be acknowledged.

7.2.1 Dataset Scope and Diversity

Environmental Constraints The dataset was recorded in a controlled laboratory environment with an empty recording space free of furniture and obstacles. This setting does not reflect realistic deployment scenarios where WiFi signals interact with complex indoor environments including walls, furniture, and other reflective surfaces. The multipath propagation characteristics in furnished homes, offices, or care facilities differ substantially from the laboratory setup, potentially limiting the generalizability of models trained exclusively on this dataset.

Limited Participant Diversity The dataset comprises recordings from one subject. This limited participant pool does not capture the full range of inter-subject variability in body dimensions, movement styles and activity execution speeds in real-world population. Models trained on this dataset may not generalize well to individuals with significantly different characteristics.

Activity Vocabulary The dataset includes ten distinct activities selected to span fine-grained and coarse-grained motion types. However, realistic monitoring scenarios involve broader activity vocabularies, including activities of daily

living (cooking, cleaning, eating), transitions between rooms, and multi-person interactions. The current activity set provides a foundation for segmentation research but does not encompass the full complexity of real-world behavior.

Single-Environment Recording All recordings were conducted in a single laboratory room with fixed transmitter and receiver positions. Cross-environment generalization — a critical requirement for practical deployment — cannot be thoroughly evaluated with this dataset. Different room geometries, sizes, and materials produce distinct CSI characteristics, and models may overfit to the specific propagation environment of the recording space.

7.2.2 Hardware and Data Collection

Hardware Configuration The dataset employs a specific hardware setup:

- **Transmitter:** Single antenna on USRP NI USRP-2954R
- **Receivers:** Three custom-mounted receivers, each with four antennas spaced 2 cm apart to produce useful phase differences for CSI sensing
- **CSI characteristics:** 114 subcarriers per antenna, sampled at 1000 Hz
- **Video:** Phone camera at 30 fps, side view
- **Motion Capture:** PhaseSpace Impulse X2E system at 960 Hz using active LED markers on a body suit and transmit/receive antenna locations

This configuration differs from typically commodity WiFi deployments in several ways. The 1000 Hz CSI sampling rate substantially exceeds the 10–100 Hz rates common in existing datasets and the capabilities of many commercial WiFi chipsets. The custom antenna spacing and USRP-based transmission provide research-grade signal quality but may not reflect the characteristics of consumer routers. Models trained on this high-rate, high-quality data may not transfer directly to lower-rate commodity hardware.

7.2.3 Benchmarking Framework Scope

The Breaking-CSI framework currently integrates three baseline segmentation models: Wi-Monitor, WiFiTAD, and TW-FINCH. While these architectures represent supervised dense classification, proposal-based detection, and unsupervised clustering respectively, many relevant model families — transformer-based temporal models, graph neural networks, contrastive self-supervised methods, and multi-modal fusion architectures — are not included. The evaluation therefore characterizes a representative cross-section of approaches rather than a comprehensive survey. Adding further models following the framework's base-class integration procedure is straightforward, but faithfully porting and verifying each model is non-trivial.

The framework computes four primary metrics (frame accuracy, mIoU, mAP at IoU 0.7, and Segmental F1 at IoU 0.5). Other metrics used in temporal segmentation research — edit distance, segmental F1 at multiple overlap thresholds, and normalized Levenshtein distance — are not currently implemented. The chosen metrics balance comprehensiveness with interpretability but do not capture every aspect of segmentation quality.

7.2.4 General CSI-Based Segmentation Limitations

Beyond limitations specific to this work, WiFi-based activity segmentation faces inherent challenges:

Environmental Sensitivity WiFi CSI encodes the full indoor propagation environment: signal paths reflect off walls, furniture, and the subject's body, so even small changes in room configuration can alter the CSI patterns associated with a given activity. A model trained in one room may not transfer reliably to another, even if the hardware setup and activities are identical. This environment-specificity has been widely recognized as a fundamental obstacle for deployment of WiFi sensing systems [67, 36, 44]. Several works have attempted to address it via domain adaptation and environment-independent feature-engineering [26, 59], but cross-environment remains an open challenge for practical deployment.

Single-Person Assumption All models evaluated in this thesis, including DopplerTAS, assume a single person is present in the sensing area. In practice, multiple people produce overlapping Doppler contributions and overlapping amplitude variations that violate this assumption. The multi-person case has been identified as key open challenge across the WiFi sensing literature [23, 2], and dedicated solutions remain an active research direction [19, 43].

Chapter 8

Conclusion and Future Work

This chapter summarizes the contributions of this thesis, answers the research questions posed in Chapter 1, and outlines directions for future work.

8.1 Summary of Contributions

This thesis addressed the infrastructure gap hindering WiFi-based activity segmentation research through three complementary contributions: a high-precision multimodal dataset, a standardized benchmarking framework, and a position-invariant segmentation model. Together, these span data collection, annotation methodology, evaluation infrastructure, and signal representation:

8.1.1 High-Precision, Multimodal Dataset

A novel CSI-based activity dataset was created with several distinguishing characteristics:

- **Motion capture-based annotation:** Leveraging synchronized PhaseSpace motion capture data enabled millisecond-accurate temporal labels, substantially exceeding the precision of manual video labeling (±33-100 ms) used in existing datasets.
- **Continuous activity sequences:** Unlike existing datasets consisting of isolated activity clips, recordings contain continuous 60-second sequences with randomized activity ordering and variable durations (3–10 s), reflecting realistic segmentation scenarios with natural transitions.
- **Data synchronization:** Precise temporal alignment of WiFi CSI (1000 Hz) and motion capture (960 Hz) enables frame-accurate activity boundary annotation and direct correlation between motion dynamics and CSI measurements.
- **Spatial diversity:** Systematic variation of recording positions across 16 floor markers prevents location-specific overfitting and encourages spatial generalization.

- **High sampling rate:** 1000 Hz CSI sampling captures fine-grained temporal dynamics, enabling research on high-rate sensing and providing a reference for studying sampling rate requirements.

The dataset comprises 210 recordings of one subject performing ten activities (five fine-grained, five coarse-grained), totaling 3.5 hours of data.

8.1.2 Annotation Infrastructure

A custom annotation tool was developed to enable precise manual labeling of activity boundaries:

- **Multi-modal display:** The tool loads synchronized video, motion capture, and CSI data for a recording and presents them in a unified interface, showing the video feed, a 3D motion capture plot, and a shared timeline simultaneously.

- **Frame-accurate manual annotation:** Annotators mark the start and end boundaries of each activity by navigating frame by frame through the timeline, allowing precise placement of labels tied directly to observed motion.

8.1.3 Unified Benchmarking Framework

A standardized evaluation framework was developed to enable fair comparison of segmentation methods:

- **Standardized data formats:** Unified input/output formats enabling plug-and-play integration of different models

- **Preprocessing pipelines:** Implementations of four CSI normalization strategies — automatic gain control (AGC), global min-max normalization, global z-score normalization, and per-packet phase detrending — configurable per model and dataset.

- **Consistent evaluation:** Fixed train-test splits, deterministic random seeds, and standardized metrics (accuracy, mIoU, mAP, F1) ensuring reproducible comparisons.

- **Model implementations:** Three baseline models integrated — Wi-Monitor (dense CNN classifier), WiFiTAD (proposal-based detection), and TW-FINCH (unsupervised clustering) — alongside the proposed DopplerTAS model, covering supervised dense classification, temporal detection, and unsupervised approaches.

- **Comprehensive logging:** Automated recording of all experimental parameters, enabling full reproducibility.

The framework reduces barriers to entry for segmentation research and enables systematic assessment of algorithmic progress.

8.1.4 DopplerTAS: A Position-Invariant Segmentation Model

To address the position-dependence of raw CSI amplitude, a dedicated temporal activity segmentation model was designed that operates on Doppler features derived from time-differential CSI phase:

- **Position-invariant representation:** By computing Doppler power spectra from the conjugate product of consecutive CSI samples, the input features capture motion-induced velocity rather than absolute channel state, making them substantially less sensitive to the subject's position in the room.
- **Multi-receiver fusion:** The three-receiver configuration stacks nine independent Doppler channels (three antennas per receiver), encoding complementary spatial perspectives on human motion.
- **Temporal modeling:** A BiLSTM processes the Doppler spectrogram sequence across a long temporal context window, enabling the model to integrate motion dynamics over time for reliable boundary detection.

8.1.5 Empirical Insights

Benchmarking three state-of-the-art segmentation models on WiPos produced several concrete findings. DopplerTAS, the proposed Doppler-informed temporal action segmentation model, achieves 96.70% frame accuracy and 90.43% mIoU on the three-receiver configuration with a window length of $T = 1500$, outperforming the strongest baseline (Wi-Monitor) by approximately 36 percentage points in accuracy. All three baselines exhibit a substantial performance drop when transferred from their native datasets to WiPos, confirming that WiPos is a genuinely challenging benchmark. Ablation experiments demonstrate that receiver count and temporal context both have large effects: adding two additional receivers increases accuracy by only 1.98 pp (94.72% $\rightarrow$ 96.70%), demonstrating that the antenna-conjugation operation, not receiver count, is the primary mechanism for position invariance; extending the window from $T = 500$ to $T = 1500$ produces a larger 20.5 pp gain, validating the importance of long temporal context as the dominant design decision in DopplerTAS.

8.2 Research Questions Answered

This section revisits the research questions posed in Section 1.2 and summarizes how this work addresses them.

RQ1: What dataset characteristics are required to support rigorous temporal activity segmentation research with WiFi CSI? Existing datasets often lack three properties critical for temporal segmentation: precise activity boundary labels, continuous multi-activity recordings, and systematic spatial diversity. This work established that millisecond-accurate boundaries require a ground-truth motion signal independent of the CSI itself; video-based labeling introduces ambiguities on the order of tens to hundreds of milliseconds that are incompatible with fine-grained boundary evaluation. Continuous recordings are necessary because isolated clip datasets do not expose models to

transition segments or variable-duration activities. Spatial diversity prevents models from exploiting position-specific CSI patterns that do not reflect the underlying activities. WiPos was designed accordingly: boundary labels are derived from motion capture marker velocities, each recording is a continuous 60-second sequence, and data were collected at 16 distinct floor positions across three receivers.

RQ2: How can a benchmarking framework provide fair and reproducible evaluation of activity segmentation models on WiFi CSI data? Fair comparison requires that all models receive identically preprocessed data, are evaluated on the same fixed splits, and are assessed with the same set of metrics. The Breaking-CSI framework achieves this through: a unified parquet-based data format that decouples preprocessing from model code; configurable, reproducible normalization pipelines (AGC, global min-max, global z-score, and per-packet phase detrending) applied identically to every model; fixed training-validation-test splits with deterministic random seeds; and a standardized four-metric evaluation suite (frame accuracy, mIoU, mAP at IoU 0.7, Segmental F1 at IoU 0.5). These design decisions eliminate the factors responsible for the incomparable results common in prior work.

RQ3: What evaluation methodology best captures the quality of temporal activity segmentation? No single metric captures all relevant aspects of segmentation quality. Frame accuracy measures per-sample correctness but is dominated by long segments and does not penalize temporal boundary errors. Mean IoU penalizes both over-segmentation and under-segmentation at the segment level and scales naturally to unequal activity durations. mAP at IoU 0.7 assesses whether individual segment proposals are precise enough to be considered correct detections, imposing a strict localization requirement. Segmental F1 combines precision and recall over matched segments and provides and aggregate measure of detection quality. Together, the four metrics expose complementary failure modes: a model may achieve high frame accuracy through a few large correct segments while performing poorly on mAP, or achieve moderate accuracy while producing well-localized boundaries reflected in high Boundary F1. The experiments confirm that these metrics produce meaningfully different rankings across models, justifying their use.

RQ4: Can Doppler features derived from CSI phase enable position-invariant activity segmentation? Yes. The benchmarking evaluation establishes that all three baselines suffer a consistent accuracy drop when evaluated on WiPos — where a subject performs activities at 16 distinct positions — compared to their native datasets, confirming that position-induced covariate shift is the dominant challenge.

DopplerTAS addresses this by precomputing Doppler spectra from the time-differential CSI phase before any network processing.

Because Doppler shifts measure radial velocity rather than absolute path length, the resulting features are substantially less sensitive to the subject's position in the room, though they are not fully position-invariant: the radial projection of a given motion changes with the subject's angle to the antenna array, so some position-dependence remains.

On the three-receiver WiPos configuration with window length $T = 1500$, DopplerTAS achieves 96.70% frame accuracy — more than 36 percentage points above the strongest baseline on WiPos. Because DopplerTAS differs from the baselines in both representation (Doppler vs. amplitude) and architecture (BiLSTM vs. ResNet-TCN or proposal-based detection), the contribution of each factor cannot be fully isolated without a controlled ablation — such as applying the Doppler representation to an amplitude-based model — which is left as future work. The results are nonetheless consistent with the hypothesis that the Doppler representation is the primary driver of the improvement on the spatially diverse WiPos dataset.

8.3 Future Work

Several directions extend this work toward more comprehensive and practical WiFi-based activity segmentation:

8.3.1 Dataset Expansion

Increased Participant Diversity Expanding the dataset to include more subjects would improve generalization and enable evaluation of inter-subject variability.

Multi-Environment Recording Collecting data across multiple rooms with varied geometries, furniture arrangements, and materials would enable assessment of cross-environment generalization-critical for practical deployment. Maintaining the same annotation precision and hardware setup across environments would provide controlled comparison.

Expanded Activity Vocabulary Incorporating activities of daily living (cooking, cleaning, eating), transitional activities (sitting down, lying down), and multi-person scenarios would increase ecological validity and application relevance.

Longer Recording Sessions Extending recording duration beyond 60 seconds to multi-minute or even hour-long sessions would better reflect continuous monitoring scenarios and enable study of long-term temporal dependencies.

8.3.2 Evaluation Methodology Refinement

Additional Metrics Incorporating metrics such as edit distance (sequence-level similarity), segmental F1 at multiple overlap thresholds, and boundary F1 would provide more nuanced evaluation of segmentation quality.

Cross-Dataset Evaluation Evaluating models across multiple CSI datasets would assess generalization beyond single-dataset performance and identify dataset-specific biases.

8.4 Closing Remarks

WiFi-based activity segmentation stands at an inflection point: the underlying sensing technology is widely deployed and the signal processing toolchain is well-understood. What the field has lacked is the research infrastructure needed to measure progress rigorously. This thesis addresses that gap along three dimensions.

The WiPos dataset provides a controlled, precisely annotated benchmark on which segmentation methods can be evaluated with millisecond-accurate labels and without the naivety of isolated-clip setups. The Breaking-CSI framework ensures that evaluations on WiPos are reproducible and comparable across models, removing the confounds that have made prior results difficult to interpret. DopplerTAS demonstrates that purpose-built architectures which respect the multipath, Doppler-rich nature of the CSI signal can substantially outperform general temporal segmentation models, achieving a 30 percentage-point improvement over the strongest baseline on the WiPos dataset.

Together, these contributions establish a foundation on which subsequent work can build: new models can be registered into Breaking-CSI and evaluated against the same splits and metrics; the dataset can be extended with additional participants or environments while maintaining the same annotation pipeline; and the empirical results provide a calibrated starting point for understanding which aspects of the problem remain unsolved. The most important of these — cross-environment generalization, multi-person scenarios, and real-time deployment — are left as open problems, but they are now open problems that can be measured.

Bibliography

[1] Zahraa S. Abdallah, Mohamed Medhat Gaber, Bala Srinivasan, and Shonali Krishnaswamy. Activity recognition with evolving data streams: A review. *ACM Comput. Surv.*, 51(4), July 2018.

[2] Fahd Abuhoureyah, Kok Swee Sim, and Yan Chiew Wong. Multi-user human activity recognition through adaptive location-independent wifi signal characteristics. *IEEE Access*, 12:112008–112024, 2024.

[3] Fadel Adib and Dina Katabi. See through walls with wifi! In *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*, pages 75–86, 2013.

[4] Iftikhar Ahmad, Arif Ullah, and Wooyeol Choi. Wifi-based human sensing with deep learning: Recent advances, challenges, and opportunities. *IEEE Open Journal of the Communications Society*, 5:3595–3623, 2024.

[5] Jesus A. Armenta-Garcia, Felix F. Gonzalez-Navarro, and Jesus Caro-Gutierrez. Wireless sensing applications with wi-fi channel state information, preprocessing techniques, and detection algorithms: A survey. *Computer Communications*, 224:254–274, 2024.

[6] Oscar Au. Origin wireless turns popular mesh routers into smart sensing stations, Dec 2025.

[7] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[8] Junjie Chang, Guanwei Guo, Jingyu Kang, Xincheng Xia, Yue Li, and Shuangshuang Feng. An efficient automatic gain control mechanism and design for wireless broadband rf receiver. In *2025 4th International Conference on Electronics, Integrated Circuits and Communication Technology (EICCT)*, pages 317–320, 2025.

[9] Liming Chen, Xiaolong Zheng, Leiyang Xu, Liang Liu, and Huadong Ma. Lightseg: An online and low-latency activity segmentation method for wi-fi sensing. In Shangguan Longfei and Priyantha Bodhi, editors, *Mobile and Ubiquitous Systems: Computing, Networking and Services*, pages 227–247, Cham, Switzerland, 2023. Springer Nature Switzerland.

[10] Tahmid Z Chowdhury. *Using Wi-Fi channel state information (CSI) for human activity recognition and fall detection.* PhD thesis, University of British Columbia, 2018.

[11] Marco Cominelli, Francesco Gringoli, and Francesco Restuccia. Exposing the csi: A systematic investigation of csi-based wi-fi sensing capabilities and limitations. In *2023 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 81–90, 2023.

[12] Guodong Ding, Fadime Sener, and Angela Yao. Temporal action segmentation: An analysis of modern techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(2):1011–1030, 2024.

[13] Rui Du, Haocheng Hua, Hailiang Xie, Xianxin Song, Zhonghao Lyu, Mengshi Hu, Narengerile, Yan Xin, Stephen McCann, Michael Montemurro, Tony Xiao Han, and Jie Xu. An overview on ieee 802.11bf: Wlan sensing. *IEEE Communications Surveys & Tutorials*, 27(1):184–217, 2025.

[14] Zexing Du, Xue Wang, Guoqing Zhou, and Qing Wang. Fast and unsupervised action boundary detection for action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3323–3332, June 2022.

[15] Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[16] Iandra Galdino, Julio C. H. Soto, Egberto Caballero, Vinicius Ferreira, Taiane Coelho Ramos, Célio Albuquerque, and Débora C. Muchaluat-Saade. ehealth csi: A wi-fi csi dataset of human activities. *IEEE Access*, 11:71003–71012, 2023.

[17] Yu Gu, Jinhai Zhan, Yusheng Ji, Jie Li, Fuji Ren, and Shangbing Gao. Mosense: An rf-based motion detection system via off-the-shelf wifi devices. *IEEE Internet of Things Journal*, 4(6):2326–2341, December 2017.

[18] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. Tool release: gathering 802.11n traces with channel state information. *SIGCOMM Comput. Commun. Rev.*, 41(1):53, January 2011.

[19] Jing He and Wei Yang. Imar: Multi-user continuous action recognition with wifi signals. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 6(3), September 2022.

[20] Ying He, Yan Chen, Yang Hu, and Bing Zeng. Wifi vision: Sensing, recognition, and detection with commodity mimo-ofdm wifi. *IEEE Internet of Things Journal*, 7(9):8296–8317, 2020.

[21] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[22] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[23] Shuokang Huang, Kaihan Li, Di You, Yichong Chen, Arvin Lin, Siying Liu, Xiaohui Li, and Julie A McCann. Wimans: A benchmark dataset for wifi-based multi-user activity sensing. In *European Conference on Computer Vision*, pages 72–91. Springer, 2024.

[24] Yuchi Ishikawa, Seito Kasai, Yoshimitsu Aoki, and Hirokatsu Kataoka. Alleviating over-segmentation errors by detecting action boundaries. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2322–2331, January 2021.

[25] Hongbo Jiang, Chao Cai, Xiaoqiang Ma, Yang Yang, and Jiangchuan Liu. Smart home based on wifi sensing: A survey. *IEEE Access*, 6:13317–13325, 2018.

[26] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, Wenyao Xu, and Lu Su. Towards environment independent device free human activity recognition. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, MobiCom '18, page 289–304, New York, NY, USA, 2018. Association for Computing Machinery.

[27] Robert M. Keenan and Le-Nam Tran. Fall detection using wi-fi signals and threshold-based activity segmentation. In *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 1–6, 2020.

[28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[29] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[30] Sateesh Kumar, Sanjay Haresh, Awais Ahmed, Andrey Konin, M. Zeeshan Zia, and Quoc-Huy Tran. Unsupervised action segmentation by joint representation learning and online clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20174–20185, June 2022.

[31] Oscar D Lara and Miguel A Labrador. A survey on human activity recognition using wearable sensors. *IEEE communications surveys & tutorials*, 15(3):1192–1209, 2013.

[32] Xiang Li, Daqing Zhang, Qin Lv, Jie Xiong, Shengjie Li, Yue Zhang, and Hong Mei. Indotrack: Device-free indoor human tracking with commodity wi-fi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):1–22, 2017.

[33] Fan Liu, Yuanhao Cui, Christos Masouros, Jie Xu, Tony Xiao Han, Yonina C. Eldar, and Stefano Buzzi. Integrated sensing and communications: Toward dual-functional wireless networks for 6g and beyond. *IEEE Journal on Selected Areas in Communications*, 40(6):1728–1767, 2022.

[34] Zhendong Liu, Le Zhang, Bing Li, Yingjie Zhou, Zhenghua Chen, and Ce Zhu. Wifi csi based temporal activity detection via dual pyramid network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 550–558, 2025.

[35] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[36] Yongsen Ma, Gang Zhou, and Shuangquan Wang. Wifi sensing with channel state information: A survey. *ACM Comput. Surv.*, 52(3), June 2019.

[37] Fabian Portner Marijn Sluijs. Breaking-CSI: A csi benchmarking framework for multiple models and multiple datasets (includes DopplerTAS). `https://gitlab.ewi.tudelft.nl/wisense/breaking-csi`, 2025.

[38] Said M. Mikki and Yahia M. M. Antar. On cross correlation in antenna arrays with applications to spatial diversity and mimo systems. *IEEE Transactions on Antennas and Propagation*, 63(4):1798–1810, 2015.

[39] Nzqo. GitHub - nzqo/usrpulse: A command server to transmit stuff with a USRP (but in rust).

[40] Sameera Palipana, David Rojas, Piyush Agrawal, and Dirk Pesch. FallDeFi: Ubiquitous Fall Detection using Commodity Wi-Fi Devices. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(4):155:1–155:25, January 2018.

[41] Fabian Portner. ESP-Sync-Blink: ESP32 firmware for hardware-synchronized LED timing reference. `https://gitlab.ewi.tudelft.nl/wisense/esp-sync-blink`, 2025.

[42] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*, MobiCom '13, page 27–38, New York, NY, USA, 2013. Association for Computing Machinery.

[43] Hamada Rizk and Ahmed Elmogy. Self-supervised wifi-based identity recognition in multi-user smart environments. *Sensors*, 25(10):3108, 2025.

[44] Siva Sai, Devansh Sharma, Mritunjay Shall Peelam, Vinay Chamola, Mohsen Guizani, and Dusit Niyato. Machine learning techniques for wi-fi csi-based recognition and sensing: A comprehensive review. *IEEE Internet of Things Journal*, pages 1–1, 2026.

[45] M Saquib Sarfraz, Naila Murray, Vivek Sharma, Ali Diba, Luc Van Gool, and Rainer Stiefelhagen. Temporally-weighted hierarchical clustering for unsupervised action segmentation. *arXiv e-prints*, pages arXiv–2103, 2021.

[46] Saquib Sarfraz, Vivek Sharma, and Rainer Stiefelhagen. Efficient parameter-free clustering using first neighbor relations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8934–8943, 2019.

[47] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

[48] Seemoo-Lab. GitHub - seemoo-lab/nexmon: The C-based Firmware Patching Framework for Broadcom/Cypress WiFi Chips that enables Monitor Mode, Frame Injection and much more.

[49] Fadime Sener and Angela Yao. Unsupervised learning and segmentation of complex activities from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[50] Marijn Sluijs. Phone-Camera-Recorder: Android camera control application and desktop cli for remote-triggered video recording. `https://github.com/MarijnSluijs/phone-camera-recorder`, 2025.

[51] Marijn Sluijs. WiseBed-Recorder: Multi-modal data recording and annotation tool for WiFi sensing. `https://gitlab.ewi.tudelft.nl/wisense/wisebed-recorder`, 2025.

[52] Naveed Tahir, Yang Liu, Tiexing Wang, Garrett E Katz, and Biao Chen. An unsupervised approach to motion detection using wifi signals. In *2023 International Conference on Machine Learning and Applications (ICMLA)*, pages 966–972. IEEE, 2023.

[53] Dazhuo Wang, Jianfei Yang, Wei Cui, Lihua Xie, and Sumei Sun. Caution: A robust wifi-based human authentication system via few-shot open-set recognition. *IEEE Internet of Things Journal*, 9(18):17323–17333, 2022.

[54] Fei Wang, Jianwei Feng, Yinliang Zhao, Xiaobin Zhang, Shiyuan Zhang, and Jinsong Han. Joint activity recognition and indoor localization with wifi fingerprints. *Ieee Access*, 7:80058–80068, 2019.

[55] Hao Wang, Daqing Zhang, Yasha Wang, Junyi Ma, Yuxiang Wang, and Shengjie Li. RT-Fall: A Real-Time and Contactless Fall Detection System with Commodity WiFi Devices. *IEEE Transactions on Mobile Computing*, 16(2):511–526, February 2017.

[56] Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 119:3–11, 2019. Deep Learning for Pattern Recognition.

[57] Wei Wang, Alex X. Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. Understanding and modeling of wifi signal based human activity recognition. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, MobiCom '15, page 65–76, New York, NY, USA, 2015. Association for Computing Machinery.

[58] Xu Wang, Linghua Zhang, Qin Cheng, and Feng Shu. Mosefi: Duration estimation robust human motion sensing via commodity wifi device. *Wireless Communications and Mobile Computing*, 2022(1):1690602, November 2022.

[59] Xu Wang, Linghua Zhang, and Feng Shu. Training-free and environment-robust human motion segmentation with commercial wifi device: An image perspective. *Applied Sciences*, 16(1):373, January 2026.

[60] Kaishun Wu, Jiang Xiao, Youwen Yi, Dihu Chen, Xiaonan Luo, and Lionel M Ni. Csi-based indoor localization. *IEEE Transactions on Parallel and Distributed Systems*, 24(7):1300–1309, 2012.

[61] Xuangou Wu, Zhaobin Chu, Panlong Yang, Chaocan Xiang, Xiao Zheng, and Wenchao Huang. Tw-see: Human activity recognition through the wall with commodity wi-fi devices. *IEEE Transactions on Vehicular Technology*, 68(1):306–319, 2019.

[62] Chunjing Xiao, Yue Lei, Yongsen Ma, Fan Zhou, and Zhiguang Qin. Deepseg: Deep-learning-based activity segmentation framework for activity recognition using wifi. *IEEE Internet of Things Journal*, 8(7):5669–5681, 2020.

[63] Jiang Xiao, Kaishun Wu, Youwen Yi, Lu Wang, and Lionel M. Ni. Fimd: Fine-grained device-free motion detection. In *2012 IEEE 18th International Conference on Parallel and Distributed Systems*, pages 229–235, Los Alamitos, CA, USA, 2012. IEEE Computer Society.

[64] Jianfei Yang, Xinyan Chen, Han Zou, Dazhuo Wang, Qianwen Xu, and Lihua Xie. Efficientfi: Toward large-scale lightweight wifi sensing via csi compression. *IEEE Internet of Things Journal*, 9(15):13086–13095, 2022.

[65] Jianfei Yang, He Huang, Yunjiao Zhou, Xinyan Chen, Yuecong Xu, Shenghai Yuan, Han Zou, Chris Xiaoxuan Lu, and Lihua Xie. Mm-fi: Multimodal non-intrusive 4d human dataset for versatile wireless sensing. *Advances in Neural Information Processing Systems*, 36:18756–18768, 2023.

[66] Xiaolong Yang, Jinglong Cheng, Xinxing Tang, and Liangbo Xie. Csi-based human behavior segmentation and recognition using commodity wi-fi. *EURASIP Journal on Wireless Communications and Networking*, 2023(1):46, June 2023.

[67] Siamak Yousefi, Hirokazu Narui, Sankalp Dayal, Stefano Ermon, and Shahrokh Valaee. A survey on behavior recognition using wifi channel state information. *IEEE Communications Magazine*, 55(10):98–104, 2017.

[68] Moustafa Youssef, Matthew Mah, and Ashok Agrawala. Challenges: device-free passive localization for wireless environments. In *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, MobiCom '07, page 222–229, New York, NY, USA, 2007. Association for Computing Machinery.

[69] Naiyu Zheng, Ruofeng Liu, Xiaoyi Fan, Cong Zhang, Lei Zhang, and Zhimeng Yin. Segall: A unified active learning framework for wireless sensing data segmentation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(3):1–27, 2025.

[70] Yue Zheng, Tianmeng Hang, Kun Qian, Chenshu Wu, Zheng Yang, and Xiancun Zhou. WiSH: The Design and Implementation of a Real-Time System for Whole-Day Human Detection. In *2017 IEEE 23rd International Conference on Parallel and Distributed Systems (ICPADS)*, pages 89–96, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720-1314, USA, December 2017. IEEE Computer Society. ISSN: 1521-9097.

[71] Shuang Zhou, Lingchao Guo, Zhaoming Lu, Xiangming Wen, and Zijun Han. Wi-monitor: Daily activity monitoring using commodity wi-fi. *IEEE Internet of Things Journal*, 10(2):1588–1604, 2022.

[72] Xianxun Zhu, Hongxuan Xu, Zhiyang Zhao, Xu Wang, Xiong Wei, Yang Zhang, and Jiancun Zuo. An environmental intrusion detection technology based on wifi. *Wireless Personal Communications*, 119(2):1425–1436, Mar 2021.

[73] Yiwei Zhuo, Hongzi Zhu, Hua Xue, and Shan Chang. Perceiving accurate csi phases with commodity wifi devices. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, 2017.

Appendix A

Motion capture marker labeling scheme

Table A.1: Motion capture marker labeling scheme. Each of the 64 LED markers is assigned a unique identifier and anatomical label corresponding to its placement on the body suit.

ID	Label	ID	Label
0	HAT_TOP	32	LEFT_HIP_FRONT_SIDE_TOP
1	HAT_RIGHT_EAR_FRONT	33	LEFT_HIP_FRONT
2	HAT_RIGHT_EYE	34	LEFT_HIP_SIDE_BOTTOM
3	HAT_LEFT_EYE	35	LEFT_UPPER_LEG_SIDE
4	HAT_LEFT_EAR_FRONT	36	LEFT_UPPER_LEG_FRONT
5	RIGHT_SHOULDER	37	LEFT_KNEE_SIDE
6	NECK_FRONT	38	LEFT_KNEE_FRONT
7	LEFT_SHOULDER	39	LEFT_ANKLE_SIDE
8	RIGHT_UPPER_ARM	40	LEFT_ANKLE_FRONT
9	RIGHT_ELBOW_TOP	41	LEFT_FOOT
10	RIGHT_ELBOW_FRONT	42	RIGHT_LOWER_LEG_SIDE
11	RIGHT_WRIST_SIDE	43	LEFT_LOWER_LEG_SIDE
12	RIGHT_WRIST_FRONT	44	HAT_LEFT_EAR_BACK
13	RIGHT_HAND	45	HAT_BACK
14	LEFT_UPPER_ARM	46	HAT_RIGHT_EAR_BACK
15	LEFT_ELBOW_TOP	47	NECK_BACK_MIDDLE
16	LEFT_ELBOW_FRONT	48	LEFT_ELBOW_BACK
17	LEFT_WRIST_SIDE	49	LEFT_WRIST_BACK
18	LEFT_WRIST_FRONT	50	RIGHT_ELBOW_BACK
19	LEFT_HAND	51	RIGHT_THIGH_UPPER
20	CHEST_UPPER	52	BACK_MIDDLE
21	CHEST_LOWER	53	BACK_LOWER
22	RIGHT_HIP_SIDE_TOP	54	LEFT_HIP_BACK
23	RIGHT_HIP_SIDE_BOTTOM	55	RIGHT_HIP_BACK
24	RIGHT_HIP_FRONT	56	LEFT_UPPER_LEG_BACK
25	RIGHT_UPPER_LEG_SIDE	57	RIGHT_UPPER_LEG_BACK
26	RIGHT_UPPER_LEG_FRONT	58	LEFT_KNEE_BACK
27	RIGHT_KNEE_SIDE	59	RIGHT_KNEE_BACK
28	RIGHT_KNEE_FRONT	60	RIGHT_ANKLE_BACK
29	RIGHT_ANKLE_SIDE	61	LEFT_ANKLE_BACK
30	RIGHT_ANKLE_FRONT	62	NECK_RIGHT_EAR
31	RIGHT_FOOT	63	NECK_LEFT_EAR

Appendix B

Dataset and Model Integration

B.1 Dataset Integration

Integrating a new dataset into the Breaking-CSI framework involves converting the dataset's original format into the standardized representation described in Section 4.2. This section briefly outlines the integration process and key considerations.

B.1.1 Porting Process

To add a new dataset, researchers implement a porting script that performs the following steps:

1. **Load original data:** Read the dataset from its original file format (e.g., '.mat', '.npy', '.csv', custom formats).
2. **Extract CSI and labels:** Extract CSI measurements and activity labels. Labels may be stored as separate files, embedded within CSI data files, or in various formats (integer indices, strings, one-hot encodings).
3. **Convert to standard shape:** Reshape CSI data to the common format `(timestamp, antennas, streams, subcarriers, 2)` where the final dimension contains amplitude and phase. Convert labels to integer indices.
4. **Save as Parquet:** Store the converted data in Apache Parquet format for efficient loading in subsequent experiments.

The framework provides base classes that define the required interface for dataset loaders, and detailed porting instructions are available in the repository documentation.

B.2 Model Integration

Integrating a new model into the framework requires adapting it to accept the standardized input format. The integration involves:

1. Modifying the model's input layer to accept `(timestamp, antennas, streams, subcarriers, 2)` shaped data
2. Implementing the framework's model base class interface
3. Defining model hyperparameters in a configuration file

Once integrated, models can be trained and evaluated on any supported dataset using the standardized pipeline. Detailed instructions are provided in the repository documentation [37].