UNIVERSITY OF VERONA

DEPARTMENT OF
Computer Science

DOCTORAL PROGRAM IN
Computer Science
XXXVIII cycle (2022-2025)

Real-Time Pre-Impact Fall Detection in Wearable Systems: Lightweight Deep Learning Architectures for Embedded Deployment

S.S.D. IINF - 05/A

Coordinator: **Prof. Umberto Castellani**

Supervisor: **Prof. Graziano Pravadelli**
Co-Supervisor: **Prof. Florenc Demrozi**

Candidate: **Muhammad Toqeer Ali**

Abstract

Falls represent a critical safety challenge, accounting for 16% of non-fatal occupational injuries and 865 workplace fatalities in the United States in 2022. Traditional post-impact fall detection systems prompt first aid after falls occur but fail to prevent injuries caused by ground impact. Pre-impact fall detection enables protective intervention before ground contact by activating safety systems, such as wearable airbags, to reduce the severity of impact. However, achieving this requires recognizing falls early enough to provide sufficient time for deploying protective devices while maintaining detection accuracy and reliability on resource-constrained wearable platforms.

This thesis addresses real-time pre-impact fall detection through three interconnected contributions advancing embedded wearable systems. First, a comprehensive dataset of 46.05 hours from 61 (+10) participants encompasses construction-specific elevation falls absent from existing benchmarks, combining controlled laboratory protocols with naturalistic construction site recordings. Integration with the KFall benchmark provides a robust foundation for model development and validation across diverse fall scenarios and operational contexts.

Second, a lightweight convolutional neural network optimized for STM32F722 microcontrollers achieves 86.69% F1-score while operating within strict memory and processing constraints. The architecture addresses critical timing requirements ignored by existing methods, but severe class imbalance produces excessive false negatives of 4.17%, limiting practical reliability for safety-critical applications.

Third, an enhanced two-stage detection pipeline addresses these limitations through generative data augmentation combined with hierarchical classification. Systematic evaluation of three generative models (Variational Autoencoder, Conditional TimeGAN (CT-GAN), and hybrid VAE-GAN) identifies effective strategies for synthesizing realistic falling patterns while preserving temporal dynamics. Beyond segment-level classification, a novel event-level aggregation stage employs Random Forest classification on consecutive segment predictions. This supervised mechanism replaces simple threshold rules, learning confidence patterns that distinguish genuine

falls from transient false positives. Validated through 5-fold subject-independent cross-validation, the complete pipeline achieves 99.5% event-level F1-score with 99.9% precision and false negatives below 1%. Post-training quantization enables deployment on embedded microcontrollers, meeting real-world timing constraints for wearable safety systems.

Real-world validation using commercial airbag safety jackets demonstrates deployment readiness. This thesis bridges academic fall detection research and deployable embedded safety systems, advancing wearable sensing technologies for occupational safety, elderly care, and real-time human activity recognition under severe resource constraints.

Keywords: Pre-impact fall detection, embedded systems, wearable sensors, convolutional neural networks, generative data augmentation, occupational safety, hierarchical classification, inertial measurement units, class imbalance, deep learning, real-time detection, Random Forest, time series classification, STM32 microcontroller, variational autoencoder, TimeGAN, subject-independent validation, construction safety.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Prof. Graziano Pravadelli, for his continuous guidance, unwavering support, and invaluable insights throughout my doctoral journey. His expertise in IoT systems and assistive technologies has been instrumental in shaping this research and fostering my development as a researcher.

I am deeply grateful to my co-supervisor, Prof. Florenc Demrozi, whose technical expertise and collaborative approach significantly enhanced the quality of this work. His contributions to the methodological development and system implementation were crucial to the success of this project, and his dedication to scientific excellence has been truly inspiring.

I extend my appreciation to the members of my annual evaluation committee for their constructive feedback and insightful suggestions, which helped me refine my research direction and improve the quality of this work at each stage of the doctoral program.

I am thankful to my international collaborator, Dr. Fadi Al Machot at the Norwegian University of Life Sciences (NMBU), for his valuable contributions and research perspectives that enriched this work through fruitful academic exchange and collaboration.

I acknowledge with gratitude the financial support provided by the Piano Nazionale di Ripresa e Resilienza (PNRR) and the industry partnership with Protechto s.r.l., which made this research possible.

Finally, my heartfelt thanks go to my wife for her steadfast support and patience during the challenging moments of this journey, and to my family and colleagues for their unwavering encouragement. Their support has been the foundation upon which this achievement rests.

Dissemination of Work

International Conferences

1. Cristian Turetta, Muhammad Toqeer Ali, Florenc Demrozi, and Graziano Pravadelli. "A Lightweight CNN for Real-Time Pre-Impact Fall Detection." 2025 Design, Automation & Test in Europe Conference (DATE). IEEE, 2025.

2. Muhammad Toqeer Ali, Cristian Turetta, Florenc Demrozi, and Graziano Pravadelli. "IMU-based pre-impact, impact, and post-impact fall detection dataset." CoMoRe-AI 2026, (Workshop at IEEE PerCom 2026).

Peer-Reviewed Journals

1. Muhammad Toqeer Ali, Cristian Turetta, Florenc Demrozi, and Graziano Pravadelli. "ICT-Based Solutions for Alzheimer's Disease Care: A Systematic Review." IEEE Access (2024).

2. Muhammad Toqeer Ali, Cristian Turetta, Florenc Demrozi, Fadi Al Machot, and Graziano Pravadelli. "Exploring Generative Data Augmentation for Real-Time Pre-Impact Fall Detection in Wearable Systems." IEEE Sensors Journal. (Under review).

Contents

List of Figures

List of Tables

List of Abbreviations

AAL Ambient Assisted Living

ADL Activities of Daily Living

AFT Approximate Free-fall Time

BLE Bluetooth Low Energy

CNN Convolutional Neural Network

DL Deep Learning

ECG Electrocardiogram

FID Fréchet Inception Distance

FP32 32-bit Floating Point

GAN Generative Adversarial Network

GRU Gated Recurrent Unit

ICT Information and Communication Technology

IMU Inertial Measurement Unit

INT8 8-bit Integer

KB Kilobyte

KL Kullback-Leibler (divergence)

LED Light Emitting Diode

LSTM Long Short-Term Memory

MCU Microcontroller Unit

MEMS Micro-Electro-Mechanical Systems

ML Machine Learning

MLP Multi-Layer Perceptron

MMD Maximum Mean Discrepancy

PCB Printed Circuit Board

RAM Random Access Memory

RBF Radial Basis Function

RF Random Forest

RGB Red Green Blue

RISC Reduced Instruction Set Computer

SD Secure Digital

SOTA State of the Art

SRAM Static Random Access Memory

VAE Variational Autoencoder

Chapter 1

Introduction

Falls pose a critical global challenge to health and occupational safety. In the United States, falls account for 16 % of non-fatal workplace injuries and caused 865 workplace fatalities in 2022 [1]. Among adults aged 65 and older, falls constitute the leading cause of injury-related deaths, resulting in approximately 3 million emergency department visits annually [2]. The economic burden exceeds $50 billion annually in the United States [2], placing strain on healthcare systems, families, and society. The global population aged 65 and older is projected to reach 1.5 billion by 2050 [3], further amplifying the urgency of effective fall prevention and detection systems. Figure 1.1 illustrates these statistics, demonstrating the scope and severity of the fall detection challenge across both occupational and elderly care domains.

Traditional fall detection systems operate reactively, identifying falls only after ground impact occurs. While these systems reduce time to emergency intervention, they cannot prevent injuries caused by the impact itself. Wearable protective devices such as airbag safety jackets offer a proactive solution by deploying protection before ground contact. However, these systems require advance warning to activate protection during the brief pre-impact phase.

This thesis develops practical solutions for real-time pre-impact fall detection in embedded wearable systems. The work advances from dataset development through lightweight neural network architectures to validated implementations suitable for occupational safety applications in construction environments. This bridges the gap between academic research and deployable embedded safety systems.

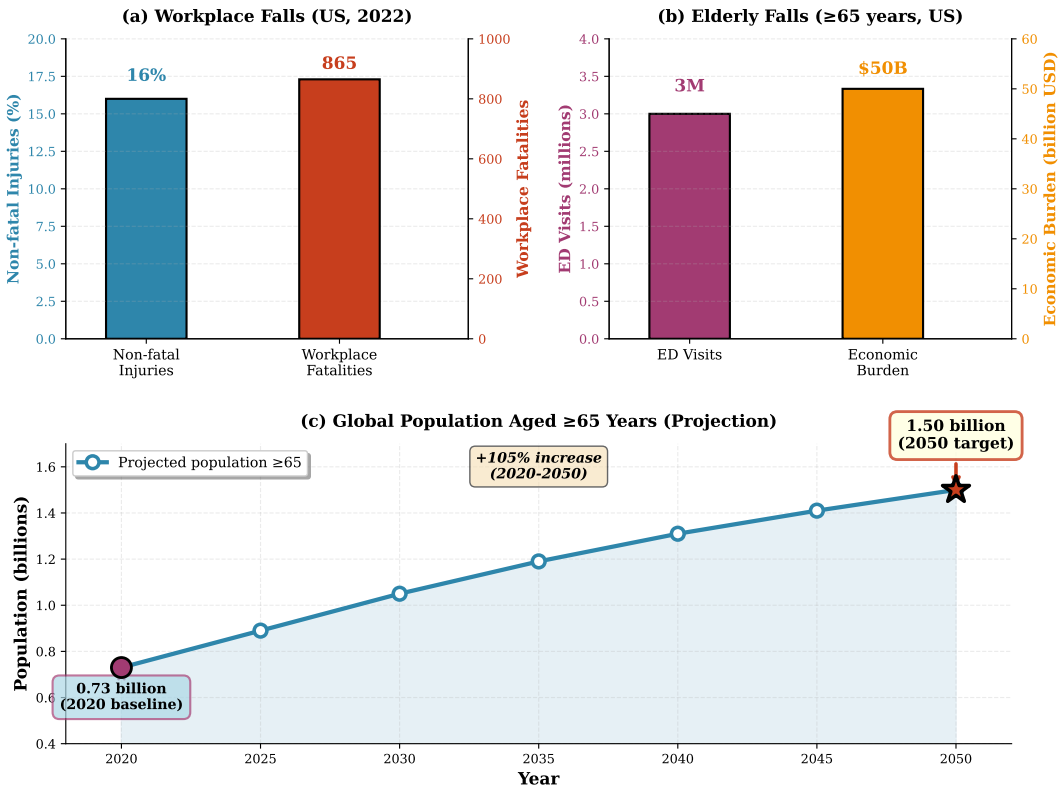


Figure 1.1. Fall-related statistics demonstrating the scope and impact of the problem. (a) Workplace fall impact in the United States (2022) showing 16% of non-fatal injuries and 865 fatalities [1]; (b) Annual elderly fall burden including 3 million emergency department visits and $50 billion economic costs [2]; (c) Projected growth in global population aged 65 and older from 0.73 billion (2020) to 1.5 billion by 2050, representing a 105% increase [3].

1.1 Research Problem and Motivation

Pre-impact fall detection presents fundamentally different challenges than conventional post-impact approaches. Detection must occur during the brief pre-impact phase using only motion patterns from fall onset, without access to distinctive ground impact signatures. Systems must distinguish genuine falls from complex daily activities while meeting strict timing requirements for protective device deployment.

1.1.1 Why Pre-Impact Fall Detection Matters

The distinction between pre-impact and post-impact detection fundamentally determines whether protective intervention can prevent injury. Post-impact systems alert responders after falls have concluded, reducing time to emergency assistance but failing to prevent impact-related injuries. In contrast, pre-impact detection enables wearable protective devices such as airbag jackets to deploy during the falling phase,

substantially reducing injury severity [4]. When an airbag system detects the onset of a fall during the 400 to 700 millisecond window before ground contact, it can inflate protective cushioning that attenuates impact forces, preventing or substantially reducing fractures, head injuries, and spinal trauma. This capability transforms fall management from reactive injury treatment into proactive injury prevention.

Quantifying this prevention potential requires understanding the injury mechanisms. Hip fractures constitute the most severe fall-related injuries in elderly populations, occurring in 5 to 10 percent of falls in individuals over 65 years and frequently resulting in permanent loss of mobility. Wearable airbag systems deployed during the pre-impact phase can reduce hip impact forces by 60 to 80 percent, substantially reducing fracture likelihood [4]. Head injuries present the second major risk. Among elderly individuals, head trauma from falls represents a leading cause of injury-related death. Pre-impact protective devices can similarly reduce head impact forces. In occupational settings, falls from elevation create even more severe consequences. Falls from heights exceeding 2 meters frequently result in multiple fractures, spinal cord injuries, and fatalities. The extended pre-impact window available in elevation falls (700 to 1000 milliseconds) provides adequate time for protective system activation.

The economic implications reinforce clinical importance. Medical costs for fatal and nonfatal falls exceeded 50 billion dollars annually in the United States as of 2018 [2]. Hip fracture treatment alone costs 40,000 to 50,000 dollars per incident, with long-term care costs substantially exceeding initial hospitalization. Preventing even a fraction of fall-related injuries through pre-impact protective devices generates substantial economic value. For construction industries with 865 workplace fatalities from falls annually, the prevention potential creates both humanitarian and economic urgency [1].

1.1.2 Current Solutions and Their Limitations

Existing fall detection research exhibits three critical limitations that prevent practical deployment in wearable safety systems. First, current solutions predominantly employ post-impact detection or ambient sensing approaches unsuitable for mobile applications requiring continuous protection regardless of location. Post-impact systems excel at identifying falls after they occur but cannot drive protective devices before impact happens. Ambient sensing approaches including camera systems and floor-mounted sensors require fixed infrastructure and do not function in outdoor

environments or mobile occupational contexts.

Second, while wearable pre-impact detection research exists, proposed approaches typically require computational resources far exceeding embedded microcontroller capabilities or rely on cloud connectivity and smartphone processing impractical for battery-powered safety devices. Research publications frequently propose sophisticated deep learning architectures optimized for desktop computers or cloud platforms. When power and size constraints require operation on microcontrollers, these architectures prove infeasible without substantial redesign. The gap between research algorithms and deployable embedded systems remains substantial.

Third, existing datasets focus primarily on controlled laboratory simulations of common daily-life falls, lacking both domain-specific scenarios such as workplace-related falls from elevation and naturalistic operational recordings essential for developing robust systems. Falls from ladders and scaffolds represent leading causes of construction fatalities yet appear in no major publicly available datasets. Similarly, real-world operational recordings capturing authentic work patterns and environmental challenges remain scarce in the research literature.

1.1.3 Research Scope and Focus

This thesis addresses pre-impact fall detection specifically for occupational safety applications in construction environments. The focus on construction contexts reflects both the severity of workplace fall hazards and the feasibility of deploying protective devices in structured work environments. Construction workers operate in controlled team settings where safety equipment receives regular maintenance and inspection, conditions less typical in elderly care applications involving independent individuals with varied technological comfort levels.

The thesis deliberately focuses on construction-specific fall scenarios including falls from elevation and equipment while incorporating general activities and falls applicable across diverse contexts. While falls among elderly individuals represent an important societal challenge, the thesis does not attempt simultaneous optimization for both construction and elderly populations. The thesis acknowledges elderly care as a complementary application domain but does not develop population-specific solutions beyond general fall detection applicable across age groups. Concentrating on construction contexts enables deeper engagement with domain-specific hazards, collaboration with industry partners for real-world validation, and development of solutions tailored to occupational safety requirements.

1.2 Research Objectives and Goals

The overarching objective of this research is to develop practical embedded pre-impact fall detection systems that operate reliably on resource-constrained wearable devices while meeting safety-critical requirements for occupational safety applications in construction environments.

This objective translates into three interconnected goals. The technical goal is to design lightweight deep learning architectures that achieve high detection accuracy within strict memory, processing, and timing constraints imposed by embedded microcontrollers, balancing computational efficiency against detection performance while enabling protective intervention before ground impact. The methodological goal is to develop comprehensive datasets capturing diverse fall scenarios including construction-specific hazards and naturalistic workplace recordings, demonstrate embedded deployment feasibility through hardware-in-the-loop validation, and establish evaluation protocols emphasizing practical reliability rather than laboratory accuracy alone. The translational goal is to establish methodologies that industry partners can adopt for commercial wearable protective devices, provide open datasets enabling reproducible research, and demonstrate that sophisticated machine learning can operate effectively within severe resource constraints when efficiency considerations guide design from initial conception.

Success is measured through quantitative performance metrics including detection accuracy and precision, false negative rates suitable for safety-critical deployment, inference latency meeting protective device activation requirements, and resource utilization within embedded microcontroller constraints. Qualitative validation through field trials and industry partner adoption provides evidence of real-world deployment readiness for occupational safety applications.

1.3 Research Questions

This thesis addresses four research questions that guide the development of embedded pre-impact fall detection systems.

RQ1: Dataset Development for Robust Pre-Impact Detection

How can comprehensive datasets encompassing diverse fall scenarios be developed to support robust pre-impact detection model training?

This question examines dataset requirements for effective pre-impact fall detection. The investigation explores data collection methodologies that capture both common fall patterns and construction-specific scenarios such as falls from elevation. The approach balances controlled laboratory protocols enabling systematic evaluation with naturalistic operational recordings capturing real workplace patterns.

RQ2: Embedded Deployment on Resource-Constrained Devices

How can pre-impact fall detection be achieved on resource-constrained embedded microcontrollers suitable for wearable devices?

This question examines deploying effective fall detection algorithms on resource-limited hardware platforms. The investigation focuses on model architectures that maintain detection accuracy while operating within strict memory, processing, and energy constraints of wearable microcontrollers.

RQ3: Addressing Class Imbalance Through Data Augmentation

How can severe class imbalance between fall and non-fall data be effectively addressed to improve detection reliability?

This question investigates overcoming data scarcity challenges in fall detection. Falls typically comprise less than 3 percent of training data, undermining model training and causing unacceptable false negative rates. The question explores advanced data augmentation strategies that synthesize realistic fall samples while preserving temporal dynamics essential for accurate detection.

RQ4: Hierarchical Classification for Enhanced Reliability

How can hierarchical classification strategies enhance fall detection performance beyond single-stage segment classification?

This question explores detection frameworks that leverage temporal context across multiple signal windows. The investigation examines whether aggregating consecutive segment predictions through supervised learning can improve detection reliability while maintaining low false positive rates essential for practical deployment.

Note that RQ3 and RQ4 are addressed together in Chapter 5, as the generative augmentation and hierarchical classification form an integrated two-stage detection pipeline.

1.4 Principal Contributions

Building on the research objectives and questions established above, this thesis makes three primary contributions that advance embedded pre-impact fall detection for wearable safety systems. Each contribution addresses critical gaps identified in existing research while demonstrating practical deployability through rigorous validation. The three technical contributions integrate peer-reviewed research papers and systematically address the four research questions through an interconnected progression from dataset development through lightweight architecture design to enhanced detection pipelines.

Contribution 1: Comprehensive Dataset with Diverse Fall Scenarios

Chapter 3 presents a comprehensive fall detection dataset that addresses RQ1 by capturing diverse fall scenarios and operational contexts. The dataset includes controlled laboratory simulations representing common falls shared across different contexts, together with construction-specific elevation falls from ladders and scaffolds and naturalistic construction site recordings. The systematic alignment methodology enables direct integration with the established KFall benchmark, providing a robust foundation for model training and subject-independent evaluation protocols. This contribution establishes the data foundation required for developing robust pre-impact detection models deployable in occupational safety applications.

Contribution 2: Lightweight CNN Architecture for Embedded Microcontrollers

Chapter 4 addresses RQ2 through a convolutional neural network architecture that achieves 86.69 percent segment-level F1-score while operating within STM32F722 microcontroller constraints. Through subject-independent cross-validation on the dataset from Contribution 1, the system demonstrates that sophisticated deep learning models can operate effectively within embedded wearable device constraints. The architecture addresses timing requirements critical for pre-impact detection, with inference completing in 4 milliseconds per window. Post-training quantization enables deployment on resource-constrained microcontrollers while preserving detection performance. This contribution establishes the feasibility of embedded deep learning

in safety-critical wearable applications.

Contribution 3: Enhanced Detection Through Generative Augmentation and Hierarchical Classification

Chapter 5 addresses both RQ3 and RQ4 through an integrated two-stage detection pipeline that reduces event-level false negatives from 4.17 percent to below 1 percent while achieving 99.9 percent precision. The approach employs systematic evaluation of generative modelling strategies (Variational Autoencoder, conditional TimeGAN, and hybrid VAE-GAN) for synthesizing realistic falling patterns while preserving temporal dynamics essential for accurate detection. This addresses severe class imbalance, where falls constitute less than 3 percent of training data. The pipeline introduces supervised event-level classification through Random Forest aggregation, learning confidence patterns from consecutive segment predictions that distinguish genuine falls from transient false positives. Independent testing by industry partner Protechto on commercial safety jackets confirms operational reliability, with construction site field trials demonstrating performance during extended work activities. This contribution completes the progression from data to deployable systems by achieving safety-critical reliability suitable for real-world occupational safety deployment.

1.5 Ethical Considerations

All research involving human participants received institutional review board approval before data collection commenced. The research adhered to established ethical principles ensuring participant safety, privacy, and autonomy throughout all experimental procedures.

Participants provided informed consent after receiving comprehensive information about research purposes, experimental procedures, potential risks, anticipated benefits, and data usage. Consent forms clearly described fall simulation procedures and associated injury risks. Participants retained the right to withdraw consent at any time without consequence. Construction site data collection employed special procedures, obtaining both individual worker consent and company authorization.

Laboratory fall simulations employed multiple protective measures minimizing injury risk. Protective equipment included safety jackets and padded clothing. Padded surfaces cushioned impacts. Trained safety supervisors provided instruction on

proper fall techniques and continuously monitored all sessions. Participant screening excluded individuals with elevated injury risk including balance disorders, recent injuries, cardiovascular conditions, or pregnancy. Gradual progression protocols introduced participants to fall procedures through demonstration, supervised practice, and incremental difficulty increases. Participants could decline any specific fall type or terminate sessions at any time.

Data anonymization procedures removed all personally identifiable information from the collected datasets. Unique anonymous identifiers prevented linkage to individual identities while enabling longitudinal analysis. Video recordings employed for ground truth annotation underwent secure storage with strictly controlled access, with deletion occurring after annotation completion. Sensor data received encryption and access controls preventing unauthorized disclosure. Construction site recordings avoided video or audio recording of workers, collecting only anonymized sensor data.

1.6 Thesis Organization

This thesis is organized into seven chapters that develop the theoretical foundations, technical contributions, and practical validation of embedded pre-impact fall detection systems for occupational safety applications.

Chapter 2: Background and Related Work reviews fall detection paradigms, sensing technologies, existing datasets, and machine learning approaches. The chapter identifies research gaps motivating the technical contributions and establishes the context for construction-specific fall detection challenges.

Chapter 3: Dataset Development and Collection addresses the first research question, presenting dataset development encompassing controlled laboratory sessions and naturalistic construction site recordings. The chapter details data collection protocols, preprocessing procedures, and integration with existing benchmarks.

Chapter 4: Lightweight CNN for Embedded Fall Detection addresses the second research question, presenting a convolutional neural network architecture optimized for microcontroller deployment. The chapter describes architecture design, training procedures, quantization strategies, hardware implementation, and validation on STM32F722 microcontrollers.

Chapter 5: Advanced Fall Detection with Generative Augmentation addresses the third and fourth research questions through enhanced detection approaches. The chapter evaluates generative models for addressing class imbalance and presents the hierarchical classification framework validated through construction site field testing.

Chapter 6: Discussion synthesizes findings across all technical contributions, providing systematic responses to research questions, critical examination of technical innovations, identification of current limitations, and discussion of promising future research directions.

Chapter 7: Conclusions summarizes the research contributions, discusses publications and academic impact, and assesses real-world deployment outcomes for construction safety applications.

Chapter 2

Background and Related Work

This chapter establishes the technical foundation for embedded pre-impact fall detection by examining fall detection paradigms, sensing technologies, datasets, and machine learning approaches. To identify relevant literature, a systematic search was conducted across academic databases including Scopus, IEEE Xplore, and Google Scholar between January 2024 and December 2024 using keywords related to fall detection, pre-impact detection, wearable sensors, deep learning, embedded systems, and construction safety. Specific search strings included: (fall detection AND wearable), (pre-impact AND fall), (fall detection AND microcontroller), (fall detection AND embedded systems), (occupational fall safety), and (construction worker safety). The search prioritized peer-reviewed articles published between 2015 and 2025, focusing on work addressing pre-impact detection using inertial sensor data with consideration for embedded hardware deployment. This search strategy, combined with forward and backward citation tracking from key publications, identified 159 papers from the initial database results. From these, 23 papers were selected as most directly relevant to pre-impact detection, embedded deployment feasibility, and construction safety applications. The review identifies critical gaps that motivate the contributions of this thesis.

Section 2.1 distinguishes pre-impact from post-impact detection paradigms and describes fall event temporal phases. Section 2.2 examines sensing technologies, focusing on wearable inertial sensors suitable for occupational safety applications. Section 2.3 evaluates existing datasets, analyzing coverage of activities, fall types, and scenarios specific to construction environments. Section 2.4 compares machine learning approaches, emphasizing suitability for embedded deployment and addressing construction-specific fall detection. Section 2.5 synthesizes identified limitations and establishes specific research gaps.

2.1 Fall Detection Paradigms

Fall detection systems employ fundamentally different operational paradigms based on when detection occurs and what intervention the system enables. This section provides technical background on these paradigms and their suitability for construction safety applications.

2.1.1 Post-Impact Detection

Post-impact detection systems identify falls through the characteristic high-magnitude acceleration spike during ground impact. This typically exceeds 3g to 6g [5]. The distinct impact signature enables straightforward detection using threshold-based algorithms or simple classifiers. Medical alert systems, commercial smartwatches, and elderly monitoring systems predominantly employ this paradigm [6, 7]. The primary advantage lies in signal clarity: impact generates unambiguous signatures easily distinguished from normal activities. However, post-impact detection cannot prevent injuries caused by the impact itself. In construction environments where falls from elevation occur, this limitation proves particularly significant. Falls from heights of 2 meters or more produce impact forces that cause severe injuries regardless of detection speed. This limitation motivates research into pre-impact approaches that enable protective intervention before ground contact.

2.1.2 Pre-Impact Detection and Fall Event Phases

Pre-impact detection identifies falling events before ground impact, enabling protective intervention through devices such as airbag systems [4]. Understanding fall event temporal structure is essential for appreciating detection system design constraints. Pre-impact detection is particularly critical in construction environments where elevation falls represent a leading cause of workplace fatalities.

Fall Event Temporal Structure. Falls from standing height progress through four distinct phases (Figure 2.1):

- **Pre-fall phase**: Normal activities before any fall-inducing event, including walking, standing, sitting, bending, and climbing. Sensors measure typical accelerations from voluntary movements and postural adjustments.
- **Falling phase**: Begins when postural control is lost. This phase typically lasts 400 to 700 milliseconds from initiation to ground impact. Approximately 50 %

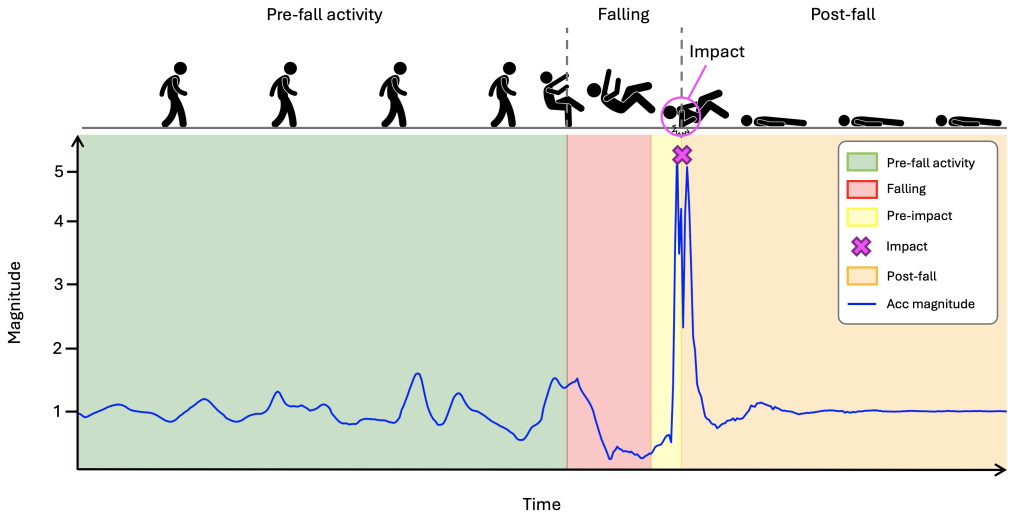


Figure 2.1. Fall event phases showing pre-fall activity (green), falling phase (red/yellow gradient), impact moment (violet cross), and post-fall phase (orange). The falling phase typically lasts 400 to 700 milliseconds from fall onset to impact, with the final 150 milliseconds reserved for protective device activation, leaving 250 to 550 milliseconds for fall detection and decision making.

of falls complete in under 500 milliseconds [4, 8]. This brief window represents the critical opportunity for pre-impact detection. In construction environments, elevation falls exhibit longer pre-impact windows (700 to 1000 milliseconds) compared to ground-level falls, providing extended time for protective system activation.

- **Impact phase**: Ground contact occurs, characterized by abrupt deceleration and high-magnitude acceleration spikes providing clear signatures for post-impact detection.

- **Post-fall phase**: Follows impact, with individuals typically remaining motionless. Some falls involve brief movement attempts before immobility [5].

Timing Constraints for Pre-Impact Systems. Figure 2.1 illustrates this temporal structure. For pre-impact systems driving protective devices, the timing budget is severely constrained. Protective devices such as wearable airbags require approximately 150 milliseconds for inflation after receiving a detection signal. This leaves only 250 to 550 milliseconds for the detection system to collect sensor data and classify the motion pattern. This constraint establishes the fundamental design requirement: detection windows must be short enough to preserve adequate time for device activation while capturing sufficient motion context for accurate classification.

Window sizes of 300 to 400 milliseconds align with these constraints, allowing detection during the falling phase while reserving time for protective intervention. In construction environments with elevation falls, the extended pre-impact window provides additional safety margin for protective system activation.

Detection Constraints and Requirements. Pre-impact systems must recognize falling events based solely on motion patterns during fall onset, without access to impact signatures. The free-fall phase exhibits characteristic acceleration patterns: reduced vertical acceleration as gravitational and inertial forces partially cancel (often below 0.5g), combined with rapid body orientation changes captured through gyroscope measurements. However, these patterns also occur during voluntary movements, creating detection ambiguity.

Systems must balance two competing requirements. Detection sensitivity ensures genuine falls are recognized, while detection specificity avoids false activations during normal activities. For wearable airbag applications in construction environments, both requirements are essential: missed detections compromise worker protection, while excessive false activations undermine practicality through nuisance deployments that erode worker confidence in safety systems.

2.2 Sensing Technologies for Fall Detection

Fall detection systems employ diverse sensing technologies, each offering distinct advantages for different deployment contexts. Wearable sensors measure body motion directly and operate independently of environmental infrastructure, making them suitable for mobile applications including construction work. Ambient sensors monitor spaces through cameras or pressure mats, providing coverage without requiring individuals to wear devices. To understand how sensing technologies are applied across assistive care applications including construction safety, this thesis draws on research literature examining diverse sensing modalities [5, 6, 7]. This section examines wearable and ambient approaches, with focus on wearable technologies most relevant for construction safety applications.

2.2.1 Wearable Inertial Sensors

Wearable inertial measurement units (IMUs) constitute the dominant approach for personal fall detection. They enable direct measurement of body motion, offer compact form factor, and operate independently from environmental infrastructure [5, 6, 7].

IMUs integrate accelerometers and gyroscopes to provide comprehensive motion capture suitable for distinguishing falls from normal construction activities.

Accelerometers measure linear acceleration along three orthogonal axes, capturing characteristic patterns during falls. During normal standing, measurements primarily reflect gravitational acceleration (approximately 1g oriented vertically). During falls, the free-fall phase exhibits reduced vertical acceleration as the body descends under gravity, often dropping below 0.5g. Subsequent ground impact generates high-magnitude spikes typically exceeding 3 to 6g. This combination of reduced acceleration followed by impact creates a distinctive pattern forming the foundation of fall detection approaches. In construction environments, tool use and equipment operation produce high-magnitude accelerations that must be distinguished from genuine fall signatures.

Gyroscopes measure angular velocity around three orthogonal axes, providing information about body rotation during falling events. Falls typically involve rapid rotational movements as the body loses balance. Peak angular velocities can exceed 200 to 300 degrees per second, substantially higher than most normal activities. Gyroscope data complements accelerometer measurements by capturing rotational dynamics not fully represented in linear acceleration. Construction workers performing normal tasks rarely achieve the rapid rotations characteristic of falling, making gyroscope measurements particularly useful for construction-specific fall detection.

Sensor placement significantly affects detection performance and wearability. Waist or lower back mounting positions sensors near the body's center of mass, providing stable measurement of whole-body dynamics while maintaining comfort. Sacral or lumbar placement captures fall-relevant motion patterns and enables integration with safety equipment such as protective jackets. Chest mounting offers stability and natural integration with protective garments. Wrist mounting benefits from existing acceptance of wrist-worn devices but captures primarily limb motion rather than whole-body dynamics. For pre-impact detection driving wearable airbags in construction environments, lower back placement provides optimal balance between measurement quality and practical integration with safety equipment.

2.2.2 Complementary Sensing Modalities

While inertial sensors form the core of wearable fall detection, other sensing modalities have been explored in broader assistive technology contexts. Physiological

sensors including heart rate monitors, blood pressure sensors, and pulse oximeters can detect sudden changes associated with falls. Environmental sensors including temperature, humidity, and pressure can characterize deployment contexts. However, these modalities address different objectives than real-time fall detection. These measurements focus on health monitoring and environmental assessment rather than motion pattern recognition required for pre-impact detection. For the specific objective of detecting falling events before ground impact in mobile wearable applications, inertial sensors remain the primary practical approach, providing the necessary motion measurements without requiring fixed infrastructure or compromising mobility essential for construction work.

2.2.3 Ambient Sensing Approaches

Ambient sensing approaches deploy sensors in the environment rather than requiring wearable devices, offering non-intrusive monitoring at the cost of requiring fixed installation [5]. These approaches prove impractical for construction environments where work occurs at multiple locations.

Camera-based systems employ computer vision to detect falls through analysis of body pose and motion patterns [5, 7]. RGB cameras provide high-resolution visual information enabling sophisticated algorithms. Depth cameras provide three-dimensional spatial information robust to lighting variations while partially addressing privacy concerns. Thermal cameras detect human presence through infrared radiation, operating in darkness while preserving privacy. However, camera systems face significant limitations including privacy concerns in private spaces, requirement for clear lines of sight, occlusion issues with furniture and equipment, and substantial computational resource requirements. In construction environments, outdoor work, equipment obstructions, and safety culture emphasizing direct employee monitoring make camera-based systems impractical.

Floor-mounted sensors including pressure-sensitive tiles detect falls through mechanical signatures. These systems offer privacy preservation and passive operation without requiring user compliance. However, they cannot distinguish which person fell in multi-occupant settings, are limited to instrumented floor areas, provide only post-impact detection, and require permanent installation. Construction sites with temporary locations and diverse flooring types cannot support fixed floor sensors.

Radio frequency sensing technologies including WiFi channel state information and millimeter-wave radar enable device-free monitoring by analyzing signal

propagation changes. However, radio-based fall detection remains primarily in the research domain with limited practical deployment due to high sensitivity to environmental changes and substantial false positive rates. Construction sites with metal equipment, vehicles, and electrical interference present hostile environments for radio-based sensing.

For construction work, ambient sensing approaches prove impractical. Construction workers operate in mobile, outdoor, and dynamic environments where fixed infrastructure cannot be maintained. Wearable sensors provide the necessary mobility and protection irrespective of work location. This thesis therefore focuses on wearable inertial sensing suitable for construction worker safety applications.

2.3 Fall Detection Datasets

To identify fall detection datasets relevant to construction safety, the datasets cited in pre-impact fall detection literature were examined. The focus was on publicly available datasets with inertial sensor data suitable for benchmarking fall detection algorithms. Specifically, datasets containing time-series IMU recordings with temporal annotations enabling pre-impact phase analysis were prioritized. This process identified four principal datasets (KFall, SisFall, MobiAct, FARSEEING) meeting these criteria and widely adopted in research. The following subsections analyze these datasets to assess their suitability for developing embedded pre-impact detection systems for construction environments.

2.3.1 KFall: The Pre-Impact Detection Benchmark

The KFall dataset represents the most widely adopted benchmark specifically designed for pre-impact fall detection [8]. The dataset contains recordings from 32 participants (16 male, 16 female, ages 22 to 40 years) performing 21 activities of daily living and 15 distinct fall types. Participants wore a single IMU sensor at the lower back, recording 6-axis inertial data at 100 Hz. The dataset provides synchronized video recordings at 100 frames per second enabling precise frame-level annotations of fall onset and impact moments.

Activities include walking at various speeds, standing, sitting, lying down, bending, squatting, jogging, jumping, and picking up objects. Fall types include forward, backward, and lateral falls, falls while walking, falls from sitting and standing positions, covering diverse scenarios representing common daily-life fall patterns.

However, KFall focuses exclusively on ground-level falls common in daily life, lacking construction-specific elevation scenarios where falls from ladders or scaffolds represent major workplace hazards.

KFall has established standard benchmark protocols widely adopted by subsequent research: 5-fold subject-independent cross-validation ensuring test participants never appear in training data, standard performance metrics computed at the segment level, and consistent evaluation procedures enabling fair comparison. These protocols provide valuable standardization for research. However, KFall lacks construction-specific scenarios and real-world work recordings essential for occupational safety applications. The participant population consists entirely of young adults who do not exhibit age-related movement characteristics or occupational experience.

2.3.2 SisFall: Multi-Age Dataset

The SisFall dataset collected data from 38 participants comprising two age groups: 19 young adults (ages 19 to 30 years) and 19 elderly individuals (ages 60 to 75 years) performing 19 activities and 15 simulated falls [9]. A key feature is sensor placement diversity, with participants wearing sensors at wrist, waist, and ankle positions. The inclusion of elderly participants provides naturalistic age-related movement patterns absent from KFall, including slower gait speeds and reduced range of motion. This diversity is valuable for understanding how fall detection works across populations.

However, SisFall exhibits limitations for construction safety applications. The dataset lacks construction-specific scenarios and real-world work recordings capturing the demanding environment of construction work. The dataset focuses on post-impact detection with many fall annotations marking the impact phase rather than fall onset. The smaller size (38 participants with 19 ADLs and 15 fall types) limits utility for training deep learning models requiring substantial data diversity. More importantly, the dataset provides limited insight into how construction workers perform normal tasks, making it difficult to assess whether detection algorithms generalize to construction environments.

2.3.3 Additional Datasets

The MobiAct dataset contains data from 66 participants (ages 20 to 47 years) performing 9 activities of daily living and 4 simulated fall types [10]. The dataset employs smartphone-based sensing with devices positioned in trouser pockets, testing feasibility of using personal smartphones for fall detection. However, the limited

activity and fall type coverage, reliance on smartphone processing, and absence of construction-specific scenarios and real-world recordings limit applicability for wearable safety systems. Smartphone placement in pockets provides less reliable whole-body dynamics measurement compared to trunk-mounted IMU sensors, particularly during construction work involving bending, reaching, and tool use.

The FARSEEING project created a unique repository of actual unplanned falls captured during daily life from elderly participants [11]. These real-world falls provide naturalistic data impossible to replicate through laboratory simulations. The dataset includes both in-home and in-facility falls representing genuine emergency situations. However, the limited size (15 participants) and uncontrolled nature make systematic model training challenging. High variability in sensor placement, unpredictable fall types, and inconsistent recording conditions complicate standardized evaluation. The dataset lacks structured ADL recordings essential for distinguishing normal activities from falls, and construction-specific scenarios remain absent. FARSEEING demonstrates the value of real-world data but highlights challenges in collecting naturalistic fall data systematically.

Dataset Implications for Construction Safety. Existing public datasets provide valuable research foundations but do not adequately represent construction work environments. Falls in construction occur on scaffolding, ladders, and elevated surfaces absent from current datasets. Construction work includes equipment operation, material handling, and interactions with tools that produce acceleration patterns unlike standard ADLs. The absence of construction-specific data creates uncertainty about whether algorithms trained on existing datasets transfer effectively to construction environments. This gap motivated the development of a new dataset incorporating construction-specific falls and naturalistic workplace recordings, presented in Chapter 3.

2.4 Machine Learning Methods for Fall Detection

Fall detection methodologies have progressed from simple threshold-based techniques through traditional machine learning to sophisticated deep learning architectures. This section systematically reviews these approaches, analyzing performance characteristics, computational requirements, and deployment constraints with emphasis on suitability for construction safety applications.

2.4.1 Threshold-Based Methods

Early research employed threshold-based methods comparing sensor measurements against fixed thresholds. These approaches detect falls when acceleration magnitude exceeds predefined thresholds or when orientation angles indicate falling. Jung et al. proposed an enhanced threshold algorithm for wearable airbags, achieving 92.40% accuracy with 96.10% recall on a subset of SisFall (30 participants, 14 ADLs, 6 fall types) [12]. The approach employs multiple threshold stages analyzing acceleration patterns and angular velocities in sequence, developing increasingly specific classification rules as data progresses through stages. De Sousa et al. developed a system incorporating subject height as an additional parameter, achieving 95.86% accuracy on combined SisFall and proprietary datasets [13]. The approach adaptively adjusts thresholds based on anthropometric measurements, recognizing that fall signatures scale with body size. Moutsis et al. created PIPTO (Precise Inertial-Based Pipeline for Threshold-Based Fall Detection), achieving 99% sensitivity on SisFall through carefully engineered multi-stage threshold analysis [14].

While computationally efficient and requiring minimal resources, threshold methods suffer from fundamental limitations. Fixed thresholds cannot adapt to diverse users, activities, and fall types, resulting in high false positive rates or missed falls. They fail to address class imbalance and lack the generalization capability of machine learning methods. In construction environments with high-magnitude accelerations from equipment and tool use, threshold-based methods require careful tuning to avoid false alarms.

2.4.2 Traditional Machine Learning Methods

Traditional machine learning (ML) methods, including support vector machines, random forests, and k-nearest neighbours, substantially improved detection accuracy compared to threshold approaches. These methods require careful feature engineering, with researchers proposing numerous features computed from accelerometer and gyroscope data, including statistical measures (mean, variance, energy), temporal characteristics (zero-crossing rate, peak count), correlation between axes, and frequency domain features.

Random forests demonstrate particular effectiveness for fall detection. The ensemble approach reduces overfitting through bootstrap aggregation and random feature subsampling, providing robustness to noisy sensor data. Random Forests offer interpretability through feature importance rankings, enabling identification of which

sensor measurements most strongly indicate falls. Multiple Random Forest variants have achieved high accuracy. Liu et al. employed Random Forests with engineered features achieving 94.5% accuracy on custom datasets combining standard ADLs with occupational tasks. Support Vector Machines with radial basis function kernels achieved 96% accuracy on KFall when trained with carefully engineered features.

Traditional ML methods achieve accuracies often exceeding 90% on benchmark datasets by learning decision boundaries separating fall and non-fall patterns through explicit feature engineering. They provide robustness to noisy data while offering interpretability valuable for safety-critical applications where understanding system behavior is important. For construction applications, the interpretability of Random Forests enables verification that decision rules reflect genuine fall characteristics rather than spurious correlations with construction equipment signatures.

However, traditional ML faces significant limitations. Heavy dependence on feature engineering creates risk that manually designed features may not capture all relevant patterns, particularly the diverse motion patterns in construction work. Computing numerous handcrafted features can impose computational costs comparable to lightweight deep learning while potentially achieving lower accuracy. Traditional approaches often fail to adequately address class imbalance and lack validation on embedded hardware platforms required for construction safety applications.

2.4.3 Deep Learning Architectures

From the systematic literature search, representative deep learning approaches for pre-impact fall detection were identified. Selection criteria included publication in peer-reviewed venues, evaluation on standard benchmarks (particularly KFall), and availability of implementation details enabling comparative analysis.

2.4.3.1 Convolutional Neural Networks

CNN-based approaches have achieved excellent results by automatically learning discriminative features from time series sensor data. The KFall dataset benchmark established ConvLSTM achieving 99.16% accuracy with 99.32% recall using 500 ms input windows on 32 participants, representing the highest accuracy reported on KFall [8]. However, the 500 ms window allocation leaves insufficient time for protective device activation given typical fall durations of 400-700 ms.

CNNs learn hierarchical representations where early layers capture low-level patterns such as signal peaks and transitions, while deeper layers learn high-level

patterns corresponding to fall characteristics. One-dimensional convolutions operating on temporal sequences slide kernels across time to detect discriminative patterns regardless of position in the input window, providing translation invariance valuable for fall detection where falls can start at any point in a monitoring window.

Chi et al. developed PreFallKD using CNN-Vision Transformer knowledge distillation, achieving 98.05% accuracy with 94.79% recall on KFall using 500 ms windows [15]. The approach transfers knowledge from a complex teacher model to a lightweight student model, enabling efficient deployment. However, the large window size remains problematic for real-time protective systems. Koo et al. developed TinyFallNet, a lightweight CNN achieving 97.97% F1-score on KFall that specifically addressed embedded deployment constraints through careful architecture design [16]. The architecture demonstrates that sophisticated performance can be achieved within embedded device constraints.

2.4.3.2 Recurrent and Hybrid Architectures

LSTM and hybrid CNN-LSTM architectures explicitly model temporal dependencies. LSTMs maintain internal hidden states across time steps, enabling them to capture long-range dependencies and temporal patterns spanning multiple seconds. Bidirectional LSTMs process sequences in both directions, leveraging future context to improve past prediction accuracy.

Jain and Semwal proposed FallNet using LSTM-CNN hybrid architecture, achieving 97.52% accuracy with 99.24% recall using 1000 ms windows on combined KFall and SisFall datasets [17]. The architecture employs CNN layers for spatial feature extraction followed by LSTM layers modeling temporal patterns. However, the 1000 ms window severely constrains practical applicability. The approach also trains on complete falling phases including impact data, creating unrealistic performance expectations. Kiran et al. developed CNN-BiGRU achieving 98.00% accuracy using 900 ms windows on KFall [18]. GRUs provide computational efficiency advantages over LSTMs through simplified gating mechanisms, though the 900 ms window leaves minimal time margin for device activation. Hnoohom et al. created FDSNeXt (Fall Detection Specialized NextNet) achieving 92.34% F1-score through deep residual networks with careful attention to temporal modeling [19].

Hybrid architectures achieve high accuracy by capturing both spatial and temporal patterns. However, their computational requirements typically exceed embedded microcontroller capabilities. Sequential processing required by recurrent architectures

prevents efficient parallelization, increasing inference time. The gating mechanisms and hidden states require substantial memory for storing intermediate activations, making deployment on memory-constrained devices challenging.

2.4.3.3 Attention and Transformer-Based Methods

Recent research has explored attention mechanisms enabling models to focus on salient temporal regions. Attention-based approaches compute relevance weights across time steps, allowing the model to identify which moments in a fall sequence are most important for classification. Chan et al. developed an attention-based CNN achieving 95.3% F1-score by learning which temporal windows within a fall sequence carry the most discriminative information. Transformer-based methods apply multi-head self-attention to capture relationships between different sensors and temporal positions. While these approaches show promise for capturing complex temporal patterns, their computational overhead makes real-time deployment on microcontrollers challenging without substantial optimization.

2.4.4 Performance Analysis and Deployment Challenges

Table 2.1 presents systematic comparison of representative approaches across all method categories. The comparison reveals clear patterns. Threshold methods offer minimal computational requirements but achieve lower accuracy (92 to 96%). Traditional machine learning improves accuracy through engineered features but lacks automatic representation learning. Deep learning approaches achieve highest accuracy (97 to 99%) and F1-scores (93 to 99%) through automatic feature learning. However, they require substantial computational resources.

Three critical limitations emerge from this analysis. First, numerous deep learning approaches employ excessively long input windows (500 to 1000 milliseconds) leaving insufficient time for protective device activation. These long windows achieve higher accuracy by capturing more context but violate fundamental deployment constraints for real-time protective systems. Second, many approaches train models on complete falling phases including impact signatures. This creates unrealistic performance expectations since impact data provides strong discriminative signals fundamentally unavailable before impact occurs in actual deployment. Third, the fundamental challenge lies in achieving high detection accuracy while meeting strict constraints of embedded wearable systems. For construction safety applications, this challenge is amplified by the need to distinguish genuine falls from high-magnitude

accelerations produced by construction equipment and tool use.

Ref.	Year	Model	Accuracy	Recall	F1-Score	WS (ms)	Dataset	# Subjects	# ADLs / # Falls
[12]	2020	Threshold algorithm	92.40	96.10	94.20	-	[9]	30	14/6
[13]	2021	Threshold algorithm	95.86	94.04	97.67	-	[9], Own	38, 9	10/5, 86/135
[8]	2021	ConvLSTM	99.16	99.32	99.01	500	[8]	32	21/15
[17]	2022	FallNet (LSTM-CNN)	97.52	99.24	98.79	1000	[8], [9]	32, 38	7/6, 7/6
[15]	2023	PreFallKD (CNN)	98.05	94.79	92.66	500	[8]	32	21/15
[18]	2024	CNN-BiGRU	98.00	98.00	98.00	900	[8]	32	21/15

Table 2.1. Recent related work on pre-impact fall detection.

2.5 Research Gaps and Motivations

The comprehensive review reveals critical technical gaps preventing practical deployment of pre-impact detection systems for wearable safety applications in construction environments. This section synthesizes these limitations and establishes specific challenges addressed by this thesis.

2.5.1 Dataset Limitations

Existing datasets exhibit three fundamental limitations that undermine the development of robust fall detection systems deployable in construction contexts. This thesis addresses these limitations through the dataset development presented in Chapter 3, which incorporates construction-specific elevation falls, naturalistic construction site recordings, and systematic integration with existing benchmarks.

Limited Scenario Coverage. Falls from ladders and scaffolds constitute major causes of construction workplace fatalities [1], yet these elevation-related fall types are absent from major benchmarks. While existing datasets capture common daily-life falls (forward, backward, lateral falls from standing or walking), they lack domain-specific scenarios representing construction hazards. Falls from elevation exhibit different pre-impact characteristics than ground-level falls, with longer pre-impact windows (700-1000 ms) and different rotational dynamics. This creates uncertainty about model generalization beyond laboratory-simulated common fall patterns.

Chapter 3 addresses this limitation by including 6 construction-specific elevation fall types representing workplace hazards absent from existing benchmarks. These falls include backward falls while moving on elevated surfaces, falls from heights, and ladder-related falls capturing the unique characteristics of construction work.

Exclusive Reliance on Laboratory Simulations. All existing datasets rely exclusively on controlled laboratory simulations. While laboratory data enables systematic evaluation, it cannot fully replicate naturalistic operational conditions in construction environments. Real-world construction work introduces sudden movements responding to unexpected obstacles, varied surface properties (metal scaffolding vs. wooden platforms), environmental factors (weather, vibration, noise), and individual behavioral patterns reflecting worker experience. Naturalistic operational recordings capturing actual deployment conditions are absent from existing resources.

Chapter 3 addresses this limitation by incorporating naturalistic construction site recordings from 10 workers, capturing movement variability during actual work activities. These recordings include normal work on scaffolding, movement on stairs, equipment operation, and other authentic construction tasks that provide the environmental context missing from laboratory protocols.

Severe Class Imbalance. Fall events constitute a small minority of movement data, creating extreme imbalance that undermines model training. Models trained on severely imbalanced data tend to bias toward predicting non-fall while failing to detect genuine falls. Most existing approaches apply simple augmentation techniques (jittering, scaling, time-warping) providing limited improvement in learning from minority class patterns. Sophisticated augmentation strategies capable of synthesizing realistic fall dynamics while preserving temporal patterns remain largely unexplored for fall detection applications.

Chapter 5 addresses this limitation through systematic evaluation of generative augmentation approaches that partially mitigate class imbalance by synthesizing additional fall samples while preserving temporal dynamics. While complete elimination of class imbalance remains challenging, these techniques demonstrate effective strategies for improving model training under severely imbalanced conditions typical of real-world deployment.

2.5.2 Methodological Limitations

Current research approaches face several critical methodological limitations that prevent practical deployment in construction environments.

Temporal Constraints Violations. Numerous approaches documented in Table 2.1 employ excessively long input windows (500 to 1000 milliseconds) leaving insufficient time for protective device activation. These long windows achieve higher

accuracy by capturing more context but violate fundamental deployment constraints. In construction contexts where falls from elevation exceed 700 ms pre-impact windows, shorter detection windows preserve safety margins essential for reliable protective system activation before ground impact.

Unrealistic Training Procedures. Several approaches in the literature train models on complete falling phases including impact signatures. This creates unrealistic performance expectations. The sharp acceleration spike during impact provides strong discriminative signals, substantially inflating accuracy metrics. However, these signals are fundamentally unavailable before impact occurs in actual deployment. Models trained with impact information cannot replicate real-world performance where only pre-impact motion patterns are available for detection.

Inadequate Class Imbalance Solutions. While severe class imbalance is widely recognized, existing solutions remain inadequate. Generative modeling approaches have demonstrated success in other time series domains including medical data augmentation and sensor-based activity recognition [20]. Yet these techniques remain largely unexplored for construction-specific fall detection. A systematic comparative evaluation of different generative architectures specifically designed to address class imbalance in construction fall detection has not been conducted.

Underdeveloped Event-Level Classification. Most research evaluates performance exclusively at the segment level, assessing accuracy on individual time windows. While segment-level performance provides essential foundation, safety-critical deployment requires understanding event-level reliability: whether complete falling events are detected before impact occurs. Simple threshold-based aggregation of segment predictions may produce excessive false negatives unsuitable for protective applications. Chapter 5 demonstrates that supervised hierarchical classification learning temporal patterns across consecutive segments substantially improves event-level detection reliability. This approach builds upon strong segment-level classification rather than replacing it, combining the strengths of automated feature learning with temporal pattern exploitation.

2.5.3 Embedded Deployment Challenges

Embedded deployment for wearable safety devices in construction environments presents substantial practical challenges that most research fails to address.

Lack of Microcontroller Validation. A substantial portion of published research evaluates algorithms on desktop computers or smartphones without validating

deployment on resource-constrained microcontrollers. Proposed architectures often contain parameters requiring memory and computational resources far exceeding typical microcontroller capabilities. When embedded deployment is considered, approaches often rely on mobile GPUs or specialized neural network accelerators unavailable in ultra-low-power wearable devices required for construction safety. For construction workers, wearable devices must operate 8+ hours per shift on battery power, eliminating power-intensive options.

Insufficient Accuracy-Efficiency Analysis. Comprehensive characterization of accuracy-efficiency trade-offs specific to pre-impact detection on microcontrollers remains limited. Detailed analysis of memory requirements, inference time, power consumption, and quantization effects is rarely provided. Hardware-in-the-loop testing on actual target microcontrollers is absent from most research. Without validation on actual deployment hardware, claimed embedded feasibility remains uncertain.

Construction-Specific Environmental Challenges. Construction sites present hostile environments for wearable sensors. Tool vibration from power equipment (drills, saws, hammers) contaminates sensor data with high-frequency noise that can trigger false alarms. Varied surfaces (metal vs. wood scaffolding, concrete floors) produce different impact characteristics. Unexpected movements from carrying heavy loads or stumbling over obstacles create acceleration profiles resembling fall onset. Temperature extremes in outdoor work affect sensor calibration. Electromagnetic interference from electrical equipment can corrupt wireless signals. Device robustness for construction work exceeds requirements for elderly care applications, requiring shock resistance and durability testing.

Field Validation Absence. Laboratory evaluation using controlled fall simulations represents essential validation but does not guarantee performance in real construction environments. While this thesis focuses primarily on laboratory validation following standard protocols, successful translation to occupational safety requires understanding real-world performance. Limited operational testing through industry collaboration provided preliminary evidence that laboratory-trained models can function in construction contexts. Comprehensive field validation with extended periods on actual construction sites remains an important direction for future work.

This comprehensive analysis of existing limitations establishes the motivation for this thesis. The following chapters address these gaps through systematic dataset development capturing diverse fall scenarios and construction-specific contexts, lightweight CNN architecture optimized for embedded deployment with microcontroller validation, and advanced detection strategies combining generative

augmentation with hierarchical classification validated through both laboratory testing and preliminary industry collaboration.

Chapter 3

Dataset Development for Pre-Impact Fall Detection

Chapter 2 identified critical limitations in existing fall detection datasets for occupational safety applications. These include the absence of construction-specific scenarios and the lack of authentic worksite recordings. This chapter presents systematic dataset development to address these gaps and answers the question: *How can comprehensive datasets encompassing diverse fall scenarios be developed to support robust pre-impact detection model training?*

The dataset combines controlled laboratory protocols with real-world construction site recordings. We enrolled 29 participants who performed 24 activities including 21 fall types. Integration with the KFall benchmark through systematic alignment procedures produces a combined resource of 61 (+10) participants suitable for robust model training and evaluation, where 61 participants performed controlled laboratory protocols and 10 additional workers contributed construction site recordings. The complete dataset, preprocessing scripts, and alignment code are publicly released to support reproducible research.

3.1 Dataset Development Rationale

Chapter 2 (Section 2.5) identified three critical limitations restricting practical deployment. These are absence of construction-specific scenarios, exclusive reliance on controlled laboratory simulations, and severe class imbalance. The developed dataset addresses these gaps through complementary strategies.

First, data collection encompasses 6 construction-specific elevation falls repre-

senting scenarios absent from existing benchmarks but constituting major causes of workplace fatalities. Second, controlled laboratory protocols are complemented with real-world construction site recordings capturing naturalistic movement patterns during actual work. Third, the dataset design follows KFall structure to enable direct integration, expanding available training data and participant diversity.

Table 3.1 positions the developed dataset relative to existing benchmarks across key dimensions.

Dataset	Subjects	ADLs	Falls	Work T.	Worksite
KFall [8]	32	21	15	No	No
SisFall [9]	38	19	15	No	No
MobiAct [10]	66	9	4	No	No
FARSEEING [11]	15	Real	Real	No	Yes
This work	**29 (+10)**	**24**	**21**	**Yes**	**Yes***

* 29 participants in controlled lab + 10 workers from construction site

Table 3.1. Comparison of fall detection datasets

3.1.1 Dataset Scope and Contributions

Data collection enrolled 29 participants who performed activities of daily living and fall simulations in controlled laboratory settings. Each participant completed a standardized session including activity performance, fall simulation, and debriefing. Fall simulations employed trained safety supervisors and protective equipment including padded surfaces and safety jackets. Participants received instruction on proper fall techniques minimizing injury risk while generating realistic fall dynamics.

The activity taxonomy includes standard ADLs overlapping with KFall plus construction-specific tasks: stair climbing with obstacles, obstacle navigation, and walking with obstacle jumping. Fall types include standard categories from KFall plus 6 construction-specific elevation falls covering backward falls while moving, falls from heights, and ladder climbing falls.

Complementing controlled protocols, real-world construction site recordings captured 10 additional workers across multiple sessions performing regular work while wearing instrumented safety jackets. These naturalistic recordings provide authentic non-fall examples representing actual deployment conditions. The complete dataset thus comprises 61 participants from controlled laboratory settings plus 10

workers from construction sites, denoted as 61 (+10) participants.

Systematic alignment with KFall through sensor configuration standardization, coordinate system transformation, and unit normalization yields a combined dataset enabling robust model development. The alignment methodology is generalizable and applicable to other publicly available datasets, enabling community-driven dataset expansion. All data, preprocessing scripts, and alignment procedures are publicly released on Zenodo.

3.2 Dataset Description

3.2.1 Participants

The study enrolled 29 healthy volunteers from a university community for controlled laboratory recordings and 10 workers from active construction sites for real-world data collection. Participant demographics are presented in Table 3.2. Laboratory participants were free from mobility impairments, musculoskeletal disorders, and cardiovascular conditions that could be exacerbated by fall simulations. All participants and workers provided written informed consent prior to data collection under institutional ethics committee approval.

	Laboratory		**Construction Site**	
Variable	**Mean ± SD**	**Range**	**Mean ± SD**	**Range**
Participants (n)	29	—	10	—
Gender (M/F)	25/6	—	10/0	—
Age (years)	23.6 ± 6.2	20–49	—	—
Height (cm)	177.6 ± 8.0	164–198	179.66 ± 6.14	170–188
Weight (kg)	71.5 ± 12.8	43–98	86.58 ± 13.93	60–115

Table 3.2. Participant demographics and anthropometric characteristics

3.2.2 Wearable Sensor System

The system architecture combines three essential components: the microcontroller for processing and decision making, inertial sensors for capturing motion data, and data storage for logging. The STM32F722RET6 microcontroller serves as the central processing unit, executing the fall detection algorithm and controlling all system

functions. This microcontroller communicates with the sensor components through an I2C interface. The microcontroller receives raw tri-axis accelerometer and gyroscope data from the LIS3DH and LSM6DS3 sensors, applies fourth-order Butterworth low-pass filtering with 5 Hz cutoff to remove high-frequency noise, computes derived features such as Euler angles and acceleration magnitudes, and feeds the processed window into the CNN inference engine. The I2C bus communication operates with minimal overhead due to the low data volume per sample. The microcontroller completes sensor data acquisition and preprocessing, then executes the CNN inference, maintaining real-time operation suitable for protective device activation. Beyond fall detection, the microcontroller additionally manages airbag system control signals, monitors battery status through analog-to-digital conversion, logs data to the microSD card, and maintains Bluetooth connectivity for remote configuration. This multi-tasking operation is coordinated through interrupt-driven design where time-critical sensor acquisition and CNN inference execute at highest priority, while data logging and communication tasks operate at lower priority levels to prevent interference with real-time detection.

The wearable sensor system integrates into a smart safety jacket designed for pre-impact fall detection in construction environments, illustrated in Figure 3.1. The jacket incorporates an airbag system triggered by the fall detection algorithm to deploy protective cushioning before ground impact. The STM32-based board embedded in the jacket includes a 6-axis Inertial Measurement Unit (IMU) sampled at 100 Hz. Following the sensor placement guidelines described in Chapter 2 (Section 2.2.1), the sensor board is positioned at the lower back to capture whole-body dynamics. The coordinate system aligns with anatomical axes: x-axis points downward, the y-axis points to the right, and the z-axis is perpendicular to the board. Table 3.3 details technical specifications.

3.2.2.1 Data Storage and Real-Time Logging

Data storage represents a critical component of the wearable system architecture. Rather than relying on wireless transmission or cloud connectivity impractical for construction work, the system implements local storage on a 32 GB microSD card integrated with the STM32F722 microcontroller through an SPI (Serial Peripheral Interface) interface. At 100 Hz sampling rate with 11 data columns (timestamp, frame counter, 3-axis accelerometer, 3-axis gyroscope, 3-axis Euler angles), the system generates approximately 12 kilobytes of sensor data per second. This continuous data

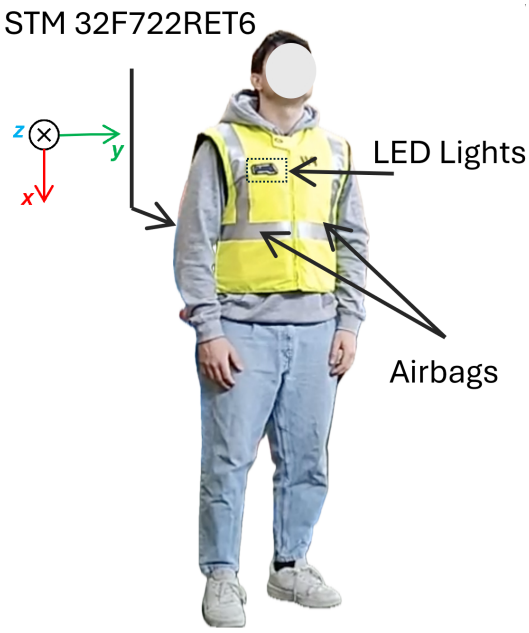


Figure 3.1. Smart safety jacket with integrated sensor system.

Feature	Specification
Microcontroller	
Model	STM32F722RET6
Processor	ARM Cortex-M7 32-bit RISC
Clock frequency	Up to 216 MHz
Flash memory	512 KB
SRAM	256 KB
Inertial Sensors	
Accelerometer	LIS3DH tri-axis MEMS
Accel. range	±16 g
Accel. resolution	1 mg
Gyroscope	LSM6DS3 tri-axis
Gyro range	±2000 dps
Gyro resolution	0.07 dps
Sampling frequency	100 Hz (both sensors)
Sensor communication	I2C interface, 400 kHz clock
Synchronization System	
LED configuration	3 high-brightness LEDs (front)
LED sequence	Red → Yellow → Green (1 s transition)
Sync marker encoding	Bit transitions: 0→4→1
Video frame rate	100 fps (synchronized)
Data Storage & Transmission	
On-board storage	32 GB microSD card (SPI interface)
Storage format	CSV files (sensor data), JSON files (metadata)
Real-time logging	Continuous data written to microSD card
Write method	DMA-assisted SPI transfers for efficient I/O
Wireless	Bluetooth 5.0 (for remote control, not data logging)
Battery life	>8 hours continuous recording
Data rate	100 samples/second × 11 columns ≈ 12 KB/second
Physical Characteristics	
PCB dimensions	6 × 4 cm
PCB weight	45 g (including battery)
Sensor placement	Lower back
Jacket integration	Embedded in rear panel

Table 3.3. Technical specifications of the wearable sensor system

stream is written directly to the microSD card in CSV format, with one row per sensor sample containing timestamp, frame counter, all six inertial measurements, and three orientation angles. The microcontroller employs DMA (Direct Memory Access) assisted transfers to write data efficiently without blocking the main processing thread, enabling simultaneous sensor acquisition, real-time fall detection processing, and continuous data logging without conflicts or data loss. Each recording session produces individual CSV files containing raw sensor measurements with precise timestamps, enabling post-hoc analysis and validation. Metadata including participant identifiers, activity task codes, and recording session parameters are stored in JSON format alongside sensor data, facilitating dataset organization and documentation. This local storage approach ensures data integrity and availability even if wireless communication becomes unavailable, a critical requirement for construction site deployments where network connectivity may be intermittent or unavailable. The 32 GB microSD card capacity supports over 700 hours of continuous recording, providing substantial storage for extended study periods and field deployments lasting multiple days.

3.2.3 Data Collection Protocol

3.2.3.1 Laboratory Sessions

Each of the 29 participants completed a standardized 45-minute laboratory session divided into four phases: orientation (5 min), activities of daily living recording (20 min), fall simulation recording (15 min), and debriefing (5 min). Participants performed 23 activities of daily living (Table 3.4) and 21 fall types (Table 3.5) in randomized order. Each activity lasted 20 to 30 seconds, with brief rest intervals between. We performed falls once per type onto 15 cm foam padding to minimize injury risk. High-speed cameras (100 fps) recorded all activities with LED synchronization markers, enabling precise temporal alignment between video frames and sensor samples.

We designed activity and fall taxonomies based on the KFall dataset structure [8], with critical additions tailored for construction safety applications. In controlled laboratory settings, 29 participants performed 23 activities (IDs 01-19, 35-36, 43-44) and 21 fall types (IDs 20-34, 37-42). Of the 23 laboratory activities, 20 overlap with KFall, while 3 construction-specific ADLs (IDs 43, 44, and later 88 from construction sites) address occupational scenarios. The 6 construction-specific elevation falls (IDs 37-42, highlighted in bold in Table 3.5) represent scenarios absent from existing

ID	Activity Description	Duration (sec)
01	Stand for 30 seconds	9–44
02	Stand, slowly bend, tie shoe lace, get up	7–22
03	Pick up an object from the floor	3–15
04	Gently jump (try to reach an object)	3–12
05	Stand, sit to ground, wait, get up with normal speed	7–23
06	Walk normally with turn	14–26
07	Walk quickly with turn	10–21
08	Jog normally with turn	9–13
09	Jog quickly with turn	3–27
10	Stumble with obstacle while walking	4–16
11	Sit on a chair for 30 seconds	18–37
12	Walk downstairs normally	7–33
13	Sit down to chair normally, get up from chair normally	8–14
14	Sit down to chair quickly, get up from chair quickly	4–11
15	Sit a moment, trying to get up, collapse into chair	6–12
16	Walk downstairs quickly	5–12
17	Lie on the floor for 30 seconds	26–40
18	Sit, lie down to floor normally, get up normally	10–22
19	Sit, lie down to floor quickly, get up quickly	8–14
35	Walk upstairs normally	9–19
36	Walk upstairs quickly	5–14
Construction-Specific Activities (Laboratory)		
43	Climb up and climb down the stairs	10–19
44	Walk slowly and jump over the obstacle	5–10
Construction-Specific Activities (Real Worksite)		
88	On field activities without falling	0.8 to 11.5 hours

Table 3.4. Activity taxonomy: daily living tasks (duration values represent mean task completion time across 29 participants, from task initiation to completion)

ID	Fall Description	Duration (sec)
20	Forward fall when trying to sit down	2–7
21	Backward fall when trying to sit down	2–5
22	Lateral fall when trying to sit down	2–5
23	Forward fall when trying to get up	2–6
24	Lateral fall when trying to get up	3–7
25	Forward fall while sitting, caused by fainting	2–4
26	Lateral fall while sitting, caused by fainting	2–6
27	Backward fall while sitting, caused by fainting	2–5
28	Vertical (forward) fall while walking caused by fainting	3–6
29	Fall while walking, using hands to dampen fall, caused by fainting	3–6
Falls While Moving		
30	Forward fall while walking caused by a trip	3–6
31	Forward fall while jogging caused by a trip	2–5
32	Forward fall while walking caused by a slip	3–6
33	Lateral fall while walking caused by a slip	3–6
34	Backward fall while walking caused by a slip	2–5
Construction-Specific Elevation Falls		
37	**Backward fall while slowly moving back**	**3–8**
38	**Backward fall while quickly moving back**	**2–4**
39	**Forward fall from height**	**3–4**
40	**Backward fall from height**	**2–6**
41	**Backward fall while climbing up the ladder**	**4–12**
42	**Backward fall while climbing down the ladder**	**3–7**

Table 3.5. Fall taxonomy: induced fall types (duration values represent mean fall event time across 29 participants, from fall onset to completion)

benchmarks yet constitute major causes of workplace fatalities.

3.2.3.2 Construction Site Recordings

Workers voluntarily wore instrumented safety jackets during normal work shifts (0.8 to 11.5 hours per session, Task ID 88). These recordings captured movements during actual tasks, including scaffold climbing, material transport, heavy lifting, and equipment operation. No falls were induced for safety reasons. Only sensor data were recorded (no video or audio) to preserve privacy, with workers identity anonymized. Including these recordings (Task 88), the dataset comprises a total of 24 activities and 21 falls.

3.2.4 Dataset Statistics

Our complete dataset comprises approximately 46.05 hours of sensor data. The controlled environment recordings from 29 participants total 3.41 hours, distributed across 23 activities (2.35 hours) and 21 fall types (1.06 hours). The actual falling time (from fall onset to ground impact) represents 4.55 minutes of the data. Construction site recordings from 10 workers contribute an additional 42.64 hours of workplace movement data (Task 88). Table 3.6 presents the recording time distribution across task categories. The dataset exhibits significant class imbalance, with falling time representing 2.22% of controlled environment data and 0.16% of the complete dataset including construction site recordings. This imbalance reflects real-world conditions where falls are rare events compared to normal activities.

Category	**Duration (hours)**	**Percentage**
Controlled Environment		
Activities (23 tasks)	2.35	5.16%
Falls (21 types)	1.06	2.33%
Construction Site		
Task 88 (workplace activities)	42.64	93.57%
Total	46.05	100%

Table 3.6. Dataset recording time distribution

The distribution of falling event durations is illustrated in Figure 3.2. Fall events demonstrate concentrated durations (mean: 476.70 ms, SD: 183.91 ms, range: 150-

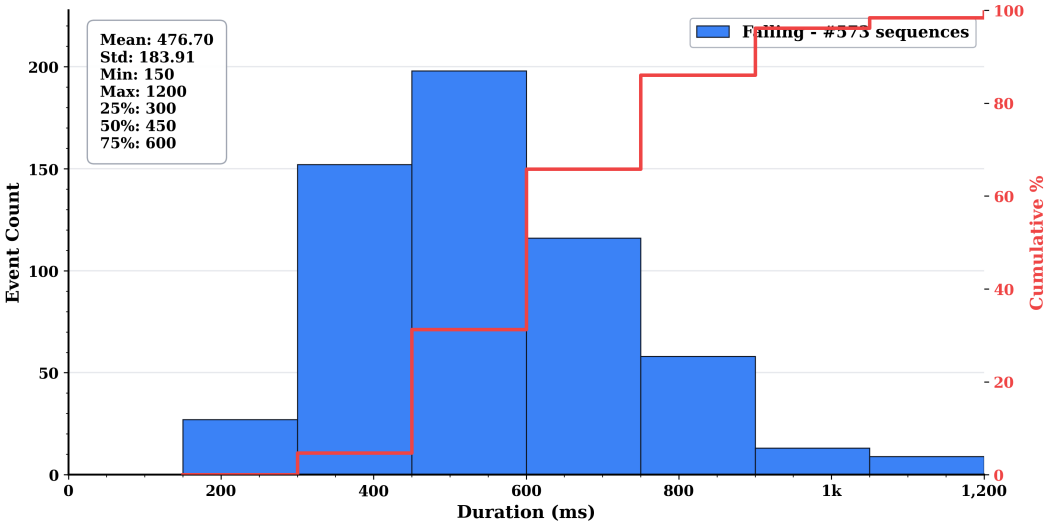


Figure 3.2. Distribution of falling event durations across 573 falls. Fall durations are concentrated around 400-550 ms (mean: 477 ms, SD: 184 ms, range: 150-1200 ms). Construction-specific elevation falls exhibit longer pre-impact phases compared to ground-level falls, providing extended reaction time for protective system activation.

1200 ms), with construction-specific elevation falls (Tasks 39-42) showing longer pre-impact phases (700-1000 ms) compared to ground-level falls (300-600 ms). This extended pre-impact window for elevation falls provides critical reaction time for activating protective systems in construction environments. The falling durations observed in laboratory settings are representative of real-world fall events, making them highly relevant for developing effective fall detection algorithms.

3.2.5 Dataset Annotation and Synchronization

Precise temporal alignment between video recordings and sensor data is critical for accurate ground truth labeling. The synchronization approach exploits LED visual markers embedded in the safety jacket.

3.2.5.1 Synchronization Protocol

At the start of each recording session, the camera and sensor data logging are initiated simultaneously via a synchronized trigger command. The three LEDs on the jacket front executed a programmed sequence: RED (500 ms), YELLOW (250 ms), GREEN (250 ms), with the transition to GREEN serving as the synchronization reference point. This LED sequence was encoded in the sensor CSV file as a dedicated synchronization channel with bit values: 0 (recording inactive), 4 (LED sequence in progress), 1

(synchronized recording active).

3.2.5.2 Data Annotation Protocol

Ground truth labels were annotated frame-by-frame using synchronized video (100 fps) and sensor data (100 Hz). For each fall event, a trained annotator used the Python video annotator tool [21] to identify two critical frames: i) fall onset, when the participant entered unrecoverable free-fall, defined as loss of postural control with downward acceleration, and ii) ground impact, when the torso touches the ground. The annotator inspected the synchronized video and sensor data frame by frame, recording these critical frame numbers. These annotations were stored in label files that contained task identifiers, subject identifiers, trial numbers, fall onset frame numbers, and impact frame numbers. These files were then mapped to precise sensor timestamps using LED synchronization markers.

Figure 3.3 illustrates an example of the annotation process, showing the temporal segmentation of a fall event into pre-fall, falling, impact, and post-fall phases based on the identified critical frames. The annotator inspected the synchronized video (i.e., 100 fps) and sensor data (i.e., 100 Hz) frame by frame, recording these critical frame numbers.

3.3 Preprocessing and Dataset Evaluation

This section describes dataset-specific preprocessing procedures and alignment methodology enabling direct integration with KFall.

3.3.1 Data Preprocessing and Alignment

3.3.1.1 Sensor Data Processing

Both the developed dataset and KFall employ 6-axis IMU data sampled at 100 Hz. Coordinate system alignment was achieved through rotation transformations, standardizing sensor orientation across datasets. We normalized measurements to consistent units: m/s² for acceleration and rad/s for angular velocity. Each IMU channel undergoes fourth-order Butterworth low-pass filtering with 5 Hz cutoff, removing high-frequency noise while preserving motion dynamics relevant for fall detection.

Figure 3.3. Temporal annotation of a fall event showing synchronized video frames and sensor data. The fall onset frame (241) marks the beginning of unrecoverable free-fall, and the impact frame (313) indicates ground contact.

3.3.1.2 Temporal Annotation and Phase Segmentation

Following the annotation methodology detailed in Subsection 3.2.5, preprocessing scripts segment each fall recording into distinct phases based on onset and impact frame numbers. Activity phases encompass all non-fall movements. This annotation structure follows the KFall protocol, ensuring compatibility between datasets.

3.3.1.3 Combined Dataset Statistics and Class Imbalance

Alignment with KFall yields a combined dataset of 61 (+10) participants: 61 participants (32 from KFall, 29 from this work) who performed controlled laboratory protocols, plus 10 additional construction workers who contributed naturalistic workplace recordings. The controlled environment data includes 23 activity tasks and 21 fall types from laboratory recordings totaling 14.51 hours (3.41 hours from our dataset plus 11.10 hours from KFall). Construction site recordings (Task 88) contribute authentic workplace activities totaling 42.64 hours. The complete combined dataset (KFall + Our) spans 57.15 hours. This distribution reflects realistic deployment scenarios where falls are rare events occurring within predominant normal activities.

3.3.1.4 Dataset Configurations

The complete dataset can be used in different configurations depending on research objectives. Throughout this thesis, we employ two configurations:

Configuration A (Controlled Environment Only): Includes all 61 participants performing laboratory activities and falls (Tasks 01-44), excluding construction site recordings (Task 88). This configuration focuses on systematic evaluation under controlled conditions with balanced representation of diverse fall types and activities. Used in Chapter 4.

Configuration B (Complete Dataset): Includes all data from Configuration A plus construction site recordings (Task 88). This provides authentic operational data representing realistic deployment conditions. The substantial construction site data increases class imbalance, reflecting real-world scenarios where falls are rare events. Used in Chapter 5.

Table 3.7 summarizes these configurations. Specific segment counts resulting from different windowing strategies are presented in the respective chapters, as window size affects the total number of segments generated through sliding window

segmentation. All subsequent chapters reference this section when discussing dataset configuration.

Characteristic	Config A	Config B
Participants	61	61 (+10)
Laboratory Data (Tasks 01-44)	✓	✓
Construction Site Data (Task 88)	×	✓
Primary Purpose	Architecture validation	Realistic evaluation
Used in Chapter	4	5

Config B: 61 lab participants + 10 construction workers

Table 3.7. Dataset configuration comparison

Per-channel z-score normalization using training set statistics standardizes sensor values, with test data normalized using training statistics to prevent data leakage.

3.3.2 Terminology and Evaluation Protocol

This section establishes the terminology and evaluation methodology used throughout this thesis.

3.3.2.1 Temporal Granularity Definitions

The thesis employs three levels of temporal granularity:

Window (or Segment): A fixed-length time window representing one model input. Throughout this thesis, the terms "window" and "segment" are used interchangeably to refer to the same concept: a sliding time window extracted from continuous sensor data. The window size varies by chapter (400 ms in Chapter 4, 300 ms in Chapter 5). Raw sensor data is divided into overlapping windows through a sliding window approach. Each window forms one input matrix for the neural network classifier. When reporting dataset statistics, we count the total number of windows (segments) generated by the sliding window process.

Event: One complete activity instance. For falls, an event spans from initiation to ground impact (typically 400 to 700 ms), encompassing multiple overlapping windows. For activities of daily living, an event represents one complete task performance (for example, one sitting action, one walk across the room). Event-level evaluation aggregates predictions across all windows within one activity instance.

Trial: One participant's performance of one activity type. Most participants performed each activity once, though some activities involved multiple repetitions recorded as separate trials.

3.3.2.2 Cross-Validation Protocol

All experiments throughout this thesis employ 5-fold subject-independent cross-validation. This ensures test participants never appear during training. The 61-participant combined dataset from controlled laboratory settings (29 participants from this work plus 32 from KFall) is divided into five folds with the following composition. For experiments employing Configuration B (Chapter 5), the 10 construction site workers are distributed across folds and contribute exclusively to the activity class, as these recordings contain no falling events.

Fold Composition: Each fold contains 12 to 15 test subjects from laboratory settings. The remaining subjects in each fold divide into approximately 42 to 45 subjects for training and 11 to 12 subjects for validation. Validation subjects are randomly selected from the training pool to provide early stopping criteria during model training. Each subject appears in the test set of exactly one fold, ensuring all subjects contribute to final performance evaluation. For Configuration B, construction site workers (subject IDs: 1001, 1002, 1003, 1004, 1005, 1006, 1007, 1008, 1109, 1110) are distributed across folds, with 7 to 9 workers in training, 2 workers in validation, and 0 to 1 worker in test sets per fold.

Subject Distribution Strategy: Subjects are distributed across folds to ensure balanced representation of demographics, activity types, and fall categories. Subject IDs in the 100-series (106-138) represent KFall participants, IDs in the 1000-series from 1009 to 1037 represent participants from this work performing laboratory protocols, and IDs 1001-1008, 1109-1110 represent construction site workers. This distribution strategy ensures each fold contains participants from all sources, preventing systematic bias toward any data source. Construction site recordings provide authentic non-fall activity data representing realistic deployment conditions.

Validation Protocol: In each fold, one subset serves as the test set, four subsets form the training pool, and validation subjects from the training pool are held out for early stopping. This protocol guarantees no participant appears in both training and test sets, providing realistic performance assessment for deployment to new users. The validation set enables early stopping based on validation loss, preventing overfitting while maximizing generalization. Chapters 4 and 5 reference this protocol without

repeating these details.

3.4 Dataset Accessibility and Usage

3.4.1 Data and Code Availability

The complete dataset, preprocessing scripts, and alignment code are publicly available on Zenodo [22] under Creative Commons Attribution 4.0 International (CC BY 4.0) license. The repository includes raw sensor data from 29 participants with frame-level annotations, construction site recordings, KFall alignment scripts, and the 5-fold cross-validation splits used throughout this thesis. The KFall dataset can be obtained from its original source [8] and integrated using the provided alignment procedures.

3.4.2 Data Format

Raw Sensor Data (*.csv*): Comma-separated files with 11 columns including timestamp (seconds), frame counter, 3-axis accelerometer (m/s^2), 3-axis gyroscope (rad/s), and 3-axis Euler angles (degrees). Sampling frequency: 100 Hz. Files are named following the convention `S[ID]T[XX]RZZ.csv` where ID represents the subject number, XX represents the task number (01-44), and ZZ indicates trial repetition. Table 3.8 provides complete specifications.

Column	Variable	Unit	Description
1	TimeStamp(s)	seconds	Sample timestamp
2	FrameCounter	integer	Sequential frame number
3	AccX	m/s^2	Lateral axis acceleration
4	AccY	m/s^2	Anterior-posterior acceleration
5	AccZ	m/s^2	Vertical axis acceleration
6	GyrX	rad/s	Roll angular velocity
7	GyrY	rad/s	Pitch angular velocity
8	GyrZ	rad/s	Yaw angular velocity
9	EulerX	degrees	Roll orientation angle
10	EulerY	degrees	Pitch orientation angle
11	EulerZ	degrees	Yaw orientation angle

Table 3.8. CSV file format specification

Annotations (*.xlsx*): Excel files containing frame-level temporal annotations for fall events. Files are named `SA[ID]_label.xlsx` where ID represents the subject

number. Table 3.9 specifies the annotation file structure.

Col.	Variable	Type	Description
1	Task Code (Task ID)	String	Fall type code and numeric ID (e.g., `F01 (20)` for fall type 20).
2	Description	String	Detailed fall event description.
3	Trial ID	Integer	Trial repetition number (always 1).
4	Fall_onset_frame	Integer	Frame number at fall initiation.
5	Fall_impact_frame	Integer	Frame number at ground contact.

Table 3.9. Annotation File Format Specification

Preprocessed Data (NPZ): Pre-segmented and filtered sensor data in NumPy compressed format, ready for direct use in machine learning frameworks.

3.4.3 Ethics and Privacy

Participants provided written informed consent for public data release. High-speed videos are not publicly released to protect participant privacy; only derived timestamp annotations are provided. Construction site recordings contain sensor data only (no video or audio) to preserve worker privacy.

3.5 Summary

This chapter developed a comprehensive dataset combining controlled laboratory protocols with authentic construction site recordings. The dataset addresses critical gaps identified in Chapter 2. These include construction-specific elevation falls absent from existing benchmarks, authentic operational recordings capturing naturalistic work patterns, and systematic alignment enabling integration with KFall for expanded training data.

The dataset provides robust foundation for model development through diverse fall scenarios and operational contexts. Precise temporal annotations with LED synchronization enable accurate pre-impact phase identification. Complete preprocessing

scripts, alignment procedures, and cross-validation protocols (Section 3.3.2) facilitate reproducible research.

Acknowledged limitations include the participant population (healthy adults aged 20 to 49), controlled laboratory protocols that may not fully capture instinctive reactions during genuine unplanned falls, and sensor placement restricted to lower back position. However, construction site recordings and KFall integration partially address these constraints through authentic operational data and diverse population representation.

Having established this comprehensive dataset foundation, Chapter 4 presents the lightweight CNN architecture that demonstrates how pre-impact detection can operate within embedded microcontroller constraints.

Chapter 4

Lightweight CNN for Embedded Fall Detection

Chapter 2 identified that existing approaches achieve high accuracy through complex architectures employing LSTM networks or transformer-based models. However, these remain impractical for deployment on resource-constrained embedded devices required for wearable safety equipment. This chapter answers the question: *How can pre-impact fall detection be achieved on resource-constrained embedded microcontrollers suitable for wearable devices?*

We present a lightweight convolutional neural network architecture specifically designed for real-time pre-impact fall detection on embedded microcontrollers. CNNs offer distinct advantages for embedded systems: parallel processing of spatial features through convolutional operations, parameter efficiency through weight sharing, and avoidance of sequential dependencies that hinder real-time operation. The methodology addresses three fundamental challenges: minimizing computational requirements to enable deployment on microcontrollers with limited flash memory and RAM, achieving sufficient detection accuracy within short temporal windows, and maintaining real-time inference speeds compatible with protective device activation requirements.

The approach employs a carefully designed CNN architecture optimized for the STM32F722 microcontroller platform. Strategic dimensionality reduction, efficient convolutional operations, and post-training quantization achieve deployment feasibility while maintaining detection performance. We evaluate the architecture through subject-independent cross-validation on the 61-participant combined dataset (controlled laboratory settings) using Configuration A (Section 3.3.1.4 of Chapter 3). Results demonstrate that sophisticated deep learning models can operate effectively

within embedded system constraints while providing reliable pre-impact fall detection suitable for driving wearable airbag systems.

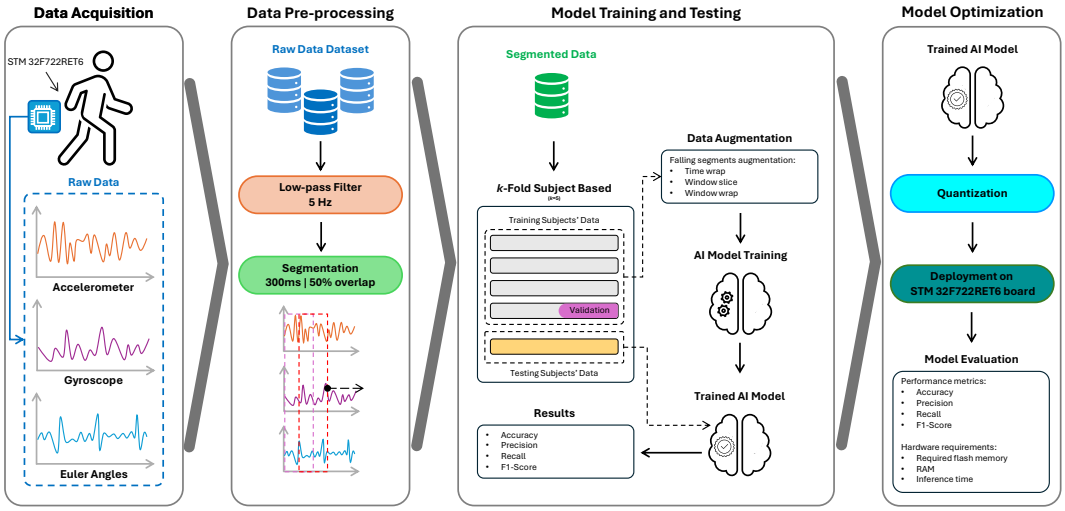


Figure 4.1. Methodological approach for lightweight CNN development, from dataset preparation through embedded deployment.

4.1 Methodology

Figure 4.1 illustrates the complete methodology for pre-impact fall recognition. Identifying pre-impact events on edge devices presents significant challenges: complex human movement patterns and limited computational resources available on embedded platforms. The methodology design prioritizes computationally efficient processes addressing these constraints.

4.1.1 Data Preprocessing

The preprocessing pipeline applies the procedures established in Chapter 3 (Section 3.3). This includes fourth-order Butterworth low-pass filtering (5 Hz cutoff), coordinate system alignment, and per-channel z-score normalization. This section focuses on the windowing strategy and temporal constraints specific to embedded deployment.

Segmentation extracts fixed-length windows for model input. Systematic experimentation with window sizes (100 to 400 ms) and overlap percentages (0 to 75 percent in 25 percent increments) identifies optimal configurations balancing temporal context against processing constraints. Sliding windows with 50 percent

overlap generate dense predictions during inference, enabling real-time operation while maintaining temporal continuity.

Each window forms a matrix with n rows representing temporal snapshots (obtained every 10 ms) and $m = 9$ columns corresponding to sensor features: tri-axial accelerometer, tri-axial gyroscope, and three Euler angles (pitch, roll, yaw). For example, a 200 ms window ($n = 20$) yields a $[20 \times 9]$ matrix. This representation preserves temporal structure essential for convolutional processing while maintaining computational efficiency suitable for embedded deployment.

4.1.2 Temporal Constraints and Window Size Selection

Following the timing constraints established in Chapter 2 (Section 2.1.2), window size selection balances three competing requirements. These are capturing sufficient temporal context for accurate fall detection, maintaining low inference latency, and preserving time for protective device activation.

Systematic experimentation with window sizes from 100 to 400 ms and overlap percentages from 0 to 75 percent identifies optimal configurations. Using 400 ms detection windows provides adequate temporal context while leaving sufficient margin for device activation in typical falls. Windows extracted from the final 150 ms before impact are excluded from training and testing to ensure the model learns recognition based solely on early-phase motion patterns available during time-critical scenarios.

CNN inference requires approximately 4 ms on the STM32F722 microcontroller, with an additional 3 ms for preprocessing. This 7 ms processing latency is negligible compared to the 400 ms data collection window. The sliding window approach with 50 percent overlap (200 ms stride) enables continuous monitoring, with new predictions generated every 200 ms.

The complete sensing and processing pipeline involves multiple latency components. Sensor acquisition latency occurs when the microcontroller retrieves data from the LIS3DH accelerometer and LSM6DS3 gyroscope through the I2C bus. The I2C bus communication with sensor hardware introduces minimal resource overhead. At 400 kHz bus speed, reading 6-axis inertial data (12 bytes) requires approximately 240 microseconds. With 100 Hz sampling frequency (10-millisecond interval), this represents 2.4 percent bus utilization. On the STM32F722 running at 216 MHz, the I2C read operation consumes approximately 48,000 CPU cycles per sample, representing 2.2 percent of available processing capacity during each sampling interval. The microcontroller implements interrupt-driven I2C transfer

using Direct Memory Access (DMA) to minimize CPU overhead. This approach allows sensor data acquisition to proceed in parallel with other processing tasks, including signal filtering, feature extraction, and CNN inference without conflicts. The remaining 97.8 percent of both bus bandwidth and CPU capacity remains available for core fall detection processing.

Sensor normalization represents a critical preprocessing step for both training and deployment. During model training on the development dataset, the mean (μ) and standard deviation (σ) are computed for each sensor channel (tri-axial accelerometer, tri-axis gyroscope, three Euler angles) across all training data. These statistics establish reference values representing typical sensor ranges and variability. The z-score normalization formula transforms each raw sensor sample: $z = (x - \mu)/\sigma$, converting raw measurements to standardized units with mean zero and standard deviation one. In deployed wearable systems, these training statistics are stored as fixed parameters within the microcontroller firmware. They are not updated during operation. Each incoming sensor sample undergoes the same z-score transformation using the pre-computed μ and σ values before being processed by the CNN. This approach maintains consistency between training and deployment by ensuring the CNN always receives sensor data in the same normalized format. Fixed normalization parameters prevent two potential problems. First, runtime statistics computation would require additional CPU cycles and memory, consuming resources needed for real-time inference. Second, continuously updating normalization parameters during operation could cause model performance drift as the CNN receives sensor data in unexpected distributions. Fixed parameters ensure reliable, consistent performance across diverse deployment environments and extended operational periods.

The 400 ms window configuration provides adequate safety margin for typical falls. However, for falls at the shorter end of the duration range, the available margin becomes constrained. The two-stage hierarchical classification approach developed in Chapter 5 addresses this limitation through optimized windowing strategies that reduce decision latency while improving detection accuracy.

Table 4.1 summarizes the complete timing budget for pre-impact fall detection with 400 ms windows.

4.1.3 Model Architecture

We propose a memory-efficient and computation-efficient deep learning solution implemented as a lightweight CNN. The CNN architecture splits the input matrix

Phase	Duration (ms)
Typical fall duration (onset to impact)	400 to 700
Detection window (data collection)	400
CNN inference + preprocessing	7
Reserved for device activation	150
Available safety margin (500 ms fall)	93
Available safety margin (700 ms fall)	293

Table 4.1. Timing budget for pre-impact fall detection with 400 ms detection windows

into three matrices, each with dimension $n \times 3$, separating the three motion features (accelerometer, gyroscope, and Euler angles) into individual processing branches.

Each motion feature matrix passes through a convolutional layer followed by a max pooling layer to extract discriminative characteristics. The three branch outputs are concatenated and fed to two dense layers comprising 64 and 32 neurons respectively, with rectified linear unit (ReLU) activation functions.

The model performs binary classification (activity versus falling) through a final dense layer with sigmoid activation. The sigmoid output is bounded between zero and one, with the prediction p indicating the confidence that a window contains falling motion.

Table 4.2 illustrates the complete architecture. The three-branch design enables parallel processing of complementary sensor modalities. Each branch applies identical convolutional and pooling operations, reducing parameters through weight sharing while maintaining the ability to learn feature representations specific to each sensor type. Concatenation of flattened branch outputs produces a combined feature vector capturing both linear acceleration and rotational dynamics. Fully connected layers then learn high-level patterns from these combined features for binary classification.

4.1.3.1 Additional Microcontroller Responsibilities

While fall detection represents the primary real-time function, the STM32F722 microcontroller manages several additional tasks. First, airbag system control requires dedicated output signals and timing-critical activation logic. When the detection pipeline identifies a fall event with sufficient confidence, the microcontroller generates a trigger signal to the airbag control electronics with precise timing to ensure deployment before ground impact. Second, battery monitoring through

Layer	Configuration	Output Shape
Input	6-channel IMU data	$[30 \times 6]$
Conv1D	64 filters, size=4, PReLU	$[27 \times 64]$
MaxPool1D	size=2, stride=2	$[13 \times 64]$
Dropout	$p = 0.1$	$[13 \times 64]$
Conv1D	64 filters, size=4, PReLU	$[10 \times 64]$
MaxPool1D	size=2, stride=2	$[5 \times 64]$
Dropout	$p = 0.4$	$[5 \times 64]$
Flatten	–	[320]
Dense	128 units, PReLU	[128]
Dropout	$p = 0.4$	[128]
Dense	2 units (logits)	[2]

Table 4.2. CNN Architecture Details.

analog-to-digital conversion tracks power levels, enabling low-battery warnings and graceful system shutdown. Third, continuous data logging to the microSD card stores all sensor readings during deployment, enabling post-incident analysis and system validation. Fourth, Bluetooth connectivity allows remote configuration of detection sensitivity, system status queries, and firmware updates without requiring physical access to the wearable device.

The microcontroller coordinates these tasks through an interrupt-driven design. The most time-critical functions (sensor acquisition at 100 Hz and CNN inference) execute at the highest priority level, guaranteeing their completion within required timing windows. Secondary tasks including data logging, Bluetooth communication, and battery monitoring, execute at lower priority levels, yielding CPU control when time-critical operations require execution. The 200 millisecond stride between consecutive detection windows provides sufficient time to complete lower-priority tasks without interfering with fall detection accuracy or timing requirements. This task hierarchy ensures reliable fall detection while enabling the comprehensive device functionality necessary for practical deployment in field environments.

4.1.4 Model Training and Evaluation

We employ subject-based 5-fold cross-validation using the 61-participant combined dataset (29 participants from Chapter 3 plus 32 from KFall). In each fold iteration, one fold containing data from approximately 12 participants serves as the test set, while the remaining four folds provide training data. Four randomly selected participants from

the training set (not used for model training) form the validation set for hyperparameter tuning and early stopping.

This methodology guarantees no overlap between training, validation, and test participants. The setup simulates realistic deployment where the fall detection model trained on one group of individuals must generalize to new, unseen users.

Terminology. Throughout this thesis, we use consistent terminology: a *window* (or *segment*) refers to a fixed-length data chunk (400 ms in this chapter) used for a single CNN prediction, while an *event* refers to a complete fall or activity comprising multiple consecutive windows.

Data Augmentation. We apply time warping and window warping augmentation techniques to falling windows in the training data. Time warping stretches and compresses the signal, making the model robust to timing variations [23]. Window warping randomly speeds up or slows down selected falling windows [24], simulating variation in overall falling event speed and producing additional falling windows relative to the original data.

Despite augmentation, the training dataset remains severely imbalanced. We address this through class weighting during training, with falling and activity classes assigned different loss weights. We set output layer bias based on the ratio of positive to negative class instances. The initial bias for class i is calculated as:

$$b_i = \log\left(\frac{p_i}{1 - p_i}\right) \tag{4.1}$$

where p_i is the prior probability of class i:

$$p_i = \frac{\text{number of instances in class } i}{\text{total number of instances}} \tag{4.2}$$

Training proceeds for up to 200 epochs with early stopping (patience of 20 epochs) based on validation loss, restoring the model weights from the best epoch. After training completes, we evaluate the model on test participant data using accuracy, precision, recall, and F1-score metrics.

4.1.5 Model Optimization

Deploying the fall detection model to drive safety device activation requires deployment on resource-constrained embedded controllers. We apply post-training quantization, which converts the model's floating-point weights and activations to lower precision format. We adopt 8-bit integer conversion after the training phase. By reducing numerical representation precision, post-training quantization significantly

decreases both model size and computational requirements. The CNN model deploys successfully on an STM32F722 microcontroller.

4.2 Experimental Results

This section evaluates lightweight CNN architecture performance through systematic experimentation. Segment-level classification results demonstrate effectiveness across multiple configurations and architectural baselines. Event-level analysis reveals practical deployment characteristics essential for safety-critical applications. Hardware deployment measurements confirm embedded feasibility on target microcontroller platforms.

4.2.1 Dataset Preparation

The evaluation employs Configuration A as defined in Section 3.3.1.4 of Chapter 3: controlled environment data only, excluding construction site recordings (Task 88). This configuration focuses on laboratory-collected activities and falls from 61 participants (29 from this work plus 32 from KFall), enabling systematic architecture validation under controlled conditions.

With 400 ms windows and 50% overlap applied to Configuration A (Section 3.3.1.4 of Chapter 3), segmentation produces 242,738 non-falling windows (96.4%) and 9,105 falling windows (3.6%), representing moderate class imbalance. All experiments follow the cross-validation protocol established in Chapter 3 (Section 3.3.2).

4.2.2 Segment-level Classification Performance

Following the evaluation terminology established in Section 3.3.2 of Chapter 3, this section evaluates performance at both segment-level and event-level granularities. Segment-level evaluation assesses the classifier's ability to correctly label individual time windows, while event-level evaluation assesses whether complete fall or activity events are correctly detected.

The lightweight CNN architecture was systematically evaluated against baseline models including Multi-Layer Perceptrons, Long Short-Term Memory networks, and ConvLSTM2D. We followed the baseline model architectures and evaluation protocols employed by Yu et al. [8]. Evaluation covered window sizes from 100 to

Model	200 ms segment size (100 ms overlap)				300 ms segment size (150 ms overlap)				400 ms segment size (200 ms overlap)			
	Accuracy	*Precision*	*Recall*	*F1-Score*	*Accuracy*	*Precision*	*Recall*	*F1-Score*	*Accuracy*	*Precision*	*Recall*	*F1-Score*
MLP	96.76	51.24	50.00	49.18	96.62	53.02	55.39	54.13	96.45	60.23	54.63	54.25
LSTM	97.28	80.92	68.62	72.98	97.43	82.51	72.08	75.93	97.60	85.97	75.74	79.81
ConvLSTM2D	97.12	81.24	61.61	66.37	97.21	83.67	63.55	68.53	97.10	85.57	65.36	70.75
CNN (Proposed)	97.93	85.61	78.85	81.75	98.01	86.38	80.03	82.85	98.28	90.40	83.95	86.69

Table 4.3. Comparison of the tested models on different segment sizes. Results are expressed in %.

400 ms and overlap configurations from 0 to 75% in 25% increments, with results presented in Table 4.3.

The proposed CNN consistently outperforms baseline architectures across all window configurations, achieving optimal performance with 400 ms windows and 50% overlap. The controlled dataset (Configuration A, Section 4.2.1 of Chapter 3) produces 242,738 non-falling windows (96.4%) and 9,105 falling windows (3.6%), representing moderate class imbalance. Despite this imbalance, the model maintains strong minority class detection while avoiding bias toward the majority class. The architecture achieves 86.69% F1-score (Table 4.3), balancing 89.32% precision with 84.26% recall.

These results reflect the realistic constraint of excluding the final 150 ms of the falling phase from training data to reserve time for airbag deployment. This differs from approaches in Table 2.1 (Chapter 2) that include the most discriminative pre-impact data but cannot meet protective device timing requirements. While those methods report higher F1-scores (92 to 99%), they cannot effectively drive safety systems as detection signals arrive too late for device activation before impact.

4.2.2.1 Event-Level Performance Analysis

Segment-level metrics provide insufficient characterization for safety-critical deployment. Each fall or activity event comprises multiple consecutive windows. For a fall event spanning 500 ms with 400 ms windows and 50% overlap, the event contains approximately 2 to 3 overlapping windows. Event-level analysis aggregates these window predictions to assess practical deployment performance.

A fall event requires correct classification of at least one window within its temporal extent for successful detection. Conversely, even a single false positive window during a non-fall activity triggers unnecessary safety device activation.

Table 4.5 presents event-level analysis results. The model achieves 4.17% event-level false negative rate, meaning 95.83% of fall events are correctly detected.

Event-level false positive rate reaches 2.04%, indicating only 2.04% of non-fall activities trigger false alarms. This low false positive rate proves critical for practical deployment, as excessive false activations undermine system usability and user acceptance. While the 4.17% false negative rate represents improvement over segment-level classification, it remains unsuitable for safety-critical deployment where missing even 4% of falls could result in serious injury. This limitation motivates the enhanced approaches in Chapter 5.

The model was deliberately configured to prioritize false positive minimization, accepting modest increases in false negatives to maintain practical viability. Activities exhibiting highest misclassification rates (Table 4.5(b)) correspond to movements rarely performed in risky contexts by target populations (construction workers, elderly individuals), further reducing practical impact of these edge cases.

Time Period	**Duration**	**Activity Events**	**False Alarms**
Single shift	8 hours	∼1,440	∼29
Full day	24 hours	∼4,320	∼88
Work week	5 days	∼21,600	∼441
Work month	22 days	∼95,040	∼1,939

Table 4.4. Projected false alarm occurrences during construction work at 2.04% event-level false positive rate

Table 4.4 illustrates the practical implications of the 2.04 percent event-level false positive rate observed in experimental validation. For an 8-hour construction work shift with continuous operation, approximately 29 false alarms would occur across 1,440 monitored activity events. This extrapolates to approximately 88 false alarms per 24 hours and approximately 441 false alarms per 5-day work week for workers wearing the system continuously. While these numbers appear substantial, they represent isolated brief detections lasting a few hundred milliseconds rather than sustained false activations. Each false alarm corresponds to a single brief window misclassification, not a prolonged erroneous alert. The construction site environment typically involves multiple workers and ongoing activity variability. The false alarm rate reflects the inherent challenge of distinguishing high-dynamic normal movements from fall-onset patterns. The mechanisms underlying false positives include vigorous activities such as rapid bending, jumping, and quick acceleration changes, which can briefly generate acceleration profiles resembling early-phase falling patterns. These false detections occur primarily during high-dynamic activities that are uncommon in safety-critical

Task ID	Miss %
39	16.00%
40	12.00%
21	9.47%
22	8.42%
41	8.00%
33	6.95%
27	5.35%
29	4.42%
37	4.00%
42	4.00%
30	3.85%
31	3.37%
32	3.17%
28	2.73%
34	2.72%
26	2.19%
23	2.17%
24	1.61%
25	1.60%
20	1.60%
38	0.00%
All actions	**4.17%**

((a)) Falls misclassified as ADLs

Task ID	Miss %
44	20.00%
15	11.29%
19	6.74%
4	6.35%
5	2.16%
10	2.13%
14	1.63%
8	1.62%
18	1.10%
9	0.56%
16	0.56%
3	0.54%
1	0.00%
2	0.00%
6	0.00%
7	0.00%
11	0.00%
12	0.00%
13	0.00%
17	0.00%
35	0.00%
36	0.00%
43	0.00%
All actions	**2.04%**
Red actions	**3.34%**
Green actions	**0.46%**

((b)) ADLs misclassified as falls

Table 4.5. Misclassification statistics (400 ms segment size). Red ADLs are unconventional for people at risk of falls due to motor impairments (e.g., elderly) or moving in risky places (e.g., at construction sites), while green ones occur more frequently.

fall contexts. Chapter 5 addresses this limitation through hierarchical classification combining generative augmentation with supervised event-level aggregation, reducing event-level false positives to 0.15 percent, corresponding to approximately 2 false alarms per 8-hour shift.

4.2.3 Embedded Deployment Performance

The CNN architecture was implemented using TensorFlow and Keras frameworks, then optimized through post-training quantization for embedded deployment. The architecture contains 89K parameters in its floating-point form. Through post-training quantization to 8-bit integer representation, the model achieves 67.03 KB size with 16.87 KB RAM usage. This fits comfortably within the STM32F722RET6 microcontroller (specifications in Table 3.3, Chapter 3).

Table 4.6 presents the quantization impact on classification performance. Results demonstrate preservation of model effectiveness with minimal degradation compared to the floating-point baseline. These results use Configuration A (controlled environment data, Section 3.3.1.4 of Chapter 3).

Metric	**FP32**	**INT8**	**Δ**
F1-score (%)	86.69	86.31	-0.38
Precision (%)	89.32	89.15	-0.17
Recall (%)	84.26	83.68	-0.58
Model Size (KB)	268.12	67.03	-75%

Table 4.6. Impact of INT8 quantization on classification performance (Configuration A)

Hardware validation on the target microcontroller platform yields inference time of 4 ms per window, with an additional 3 ms for sensor data preprocessing. This 7 ms total processing time is negligible compared to the 400 ms window duration and 200 ms stride between consecutive windows. The system maintains real-time operation with sufficient margin for protective device activation, validating the architecture's suitability for safety-critical deployment.

4.3 Summary and Transition

This chapter demonstrated that lightweight CNN architectures optimized for embedded microcontrollers can maintain sufficient pre-impact fall detection accuracy. The methodology achieves three key results.

First, careful architectural design with microcontroller constraints as primary considerations enables sophisticated deep learning deployment without extensive computational resources. The lightweight CNN achieves 86.69% segment-level F1-

score while operating within stringent embedded constraints: 67 KB model size, 17 KB RAM usage, and 4 ms inference time on STM32F722 microcontrollers.

Second, subject-independent cross-validation across 61 participants demonstrates robust generalization essential for practical deployment across diverse populations. The model achieves event-level false negative rate of 4.17% and false positive rate of 2.04%, with particularly low false alarm rates critical for user acceptance.

Third, hardware validation confirms real-time operation feasibility within temporal constraints required for protective device activation. The 400 ms detection windows with 150 ms safety margin preserve adequate time for wearable airbag deployment in typical 500 to 700 ms falls.

However, the analysis reveals limitations requiring further investigation. The event-level false negative rate of 4.17% remains unsuitable for safety-critical applications where missing even a small fraction of falls could result in serious injury. Severe class imbalance with falling windows representing only 3.6% of data limits training effectiveness despite conventional augmentation strategies. Furthermore, segment-level classification does not fully exploit temporal patterns across consecutive predictions that could enhance detection reliability.

These observations motivate Chapter 5, which addresses two complementary approaches. Advanced generative models may address class imbalance by synthesizing realistic falling patterns while preserving temporal dynamics. Hierarchical classification strategies combining segment-level predictions with event-level aggregation may reduce false negatives through temporal consistency while maintaining low false positive rates. The next chapter demonstrates that combining these approaches reduces false negatives to below 1% while maintaining 99.9% precision.

Chapter 5

Advanced Fall Detection with Generative Augmentation

Chapter 4 demonstrated successful embedded deployment of a lightweight CNN achieving 86.69% segment-level F1-score on STM32F722 microcontrollers. The architecture addressed critical timing constraints while operating within embedded resource limits. However, three fundamental limitations remained: severe class imbalance persisted despite conventional augmentation techniques, event-level false negative rate of 4.17% caused missed detections in safety-critical scenarios, and reliance on simple threshold-based aggregation rather than supervised event-level classification. Building on that foundation, this chapter answers two questions: *How can severe class imbalance between fall and non-fall data be effectively addressed to improve detection reliability? How can hierarchical classification strategies enhance fall detection performance beyond single-stage segment classification?*

This chapter presents an enhanced two-stage detection pipeline combining generative data augmentation with hierarchical classification. Three generative approaches are systematically evaluated: Variational Autoencoders (VAE), conditional TimeGAN (CT-GAN), and hybrid VAE-GAN architectures. These models synthesize realistic falling patterns while preserving temporal dynamics critical for accurate detection. Beyond segment-level classification, a novel event-level aggregation stage employs Random Forest classification on consecutive segment predictions. This replaces simple threshold rules with supervised learning that captures confidence patterns distinguishing genuine falls from transient false positives.

The methodology extends the lightweight CNN from Chapter 4 using the 61 (+10) participant combined dataset from Chapter 3: 61 participants from controlled laboratory settings plus 10 construction workers who contributed naturalistic work-

place recordings. It maintains deployment feasibility on embedded microcontrollers while improving detection reliability. Unlike Chapter 4 which employed 400 ms windows on Configuration A, this chapter adopts 300 ms windows on Configuration B (Section 3.3.1.4 of Chapter 3). The shorter windows enable more efficient two-stage operation, while Configuration B provides realistic evaluation including construction site recordings. We validate using the cross-validation protocol from Chapter 3 (Section 3.3.2). Results demonstrate that the complete pipeline achieves 99.50% event-level F1-score with false negatives reduced to below 1% and precision increased to 99.9%, meeting stringent requirements for safety-critical deployment.

5.1 Motivation

Chapter 4 demonstrated successful embedded deployment of a lightweight CNN achieving 86.69% segment-level F1-score on STM32F722 microcontrollers. The architecture addressed critical timing constraints while operating within embedded resource limits. However, three fundamental limitations remained: severe class imbalance persisted despite conventional augmentation techniques (time warping and window warping), event-level false negative rate of 4.17% caused missed detections in safety-critical scenarios, and reliance on simple threshold-based aggregation rather than supervised event-level classification.

Table 5.1 identifies four open challenges in existing approaches. The lightweight CNN addressed challenges C1 (timing constraints) and C4 (embedded validation). This chapter addresses the remaining challenges: **C2** accuracy requirements for distinguishing genuine falls from high-dynamic activities while maintaining high sensitivity, and **C3** severe class imbalance inadequately addressed by conventional augmentation.

Two complementary strategies address these challenges. Advanced generative data augmentation using deep learning models (VAE, CT-GAN, and hybrid approaches) synthesizes additional falling samples. This goes beyond conventional augmentation by generating entirely new realistic fall patterns while preserving temporal coherence. Hierarchical classification introduces a two-stage detection pipeline where Random Forest performs supervised event-level aggregation on consecutive CNN predictions. This learns confidence patterns that reduce false negatives from 4.17% to below 1% while maintaining 99.9% precision.

Ref.	Year	Model	Acc. (%)	Rec. (%)	F1 (%)	WS (ms)	Dataset	# Sub.	# ADLs/Falls	Open Challenges
[12]	2020	Threshold	92.40	96.10	94.20	-	[9]	30	14/6	C1, C2, C3
[8]	2021	ConvLSTM	99.16	99.32	99.01	500	[8]	32	21/15	C1, C2, C3, C4
[13]	2022	Threshold	95.86	94.04	97.67	-	[9], Own	38, 9	10/5, 86/135	C1, C3, C4
[17]	2022	LSTM-CNN	97.52	99.24	98.79	1000	[8], [9]	32, 38	7/6, 7/6	C1, C2, C3, C4
[19]	2023	FDSNeXt	91.87	89.92	92.34	500	[8]	32	21/15	C1, C3
[15]	2023	PreFallKD	98.05	94.79	92.66	500	[8]	32	21/15	C1, C2, C3
[16]	2023	TinyFallNet	98.00	99.32	97.97	500	[8], [11]	32, 10	21/15, Real	C1, C3
[18]	2024	CNN-BiGRU	98.00	98.00	98.00	900	[8]	32	21/15	C1, C2, C4
SOTA [4]	2025	CNN	98.28	83.95	86.69	400	[8], Own	32, 29	21/15, 23/21	C2, C3

WS: Window Size; # Sub.: Number of subjects; Acc.: Accuracy; Rec.: Recall; F1: F1-Score.

Table 5.1. Comparison of pre-impact fall detection methods addressing four open challenges: C1 (timing constraints), C2 (accuracy requirements), C3 (class imbalance), C4 (embedded deployment).

5.2 Dataset and Preprocessing

This chapter employs Configuration B (Section 3.3.1.4 of Chapter 3). This is the complete 61 (+10) participant combined dataset: 61 participants from controlled laboratory settings (29 from this work plus 32 from KFall) plus 10 construction workers contributing naturalistic workplace recordings (Task 88). Configuration B enables evaluation under realistic deployment conditions with naturalistic operational data. This differs from Chapter 4 which employed Configuration A.

Table 5.2 presents the dataset composition with this important distinction between fall recording session duration and actual falling phase duration. The severe imbalance (131:1 ratio) motivates the generative augmentation strategies presented in Section 5.3.2.

Dataset	Subjects	Activities (min)	Fall Sessions (min)	Falling Phase (min)	Total (min)
KFall	32	378.00	287.40	19.97	665.40
Our Dataset	29	141.00	63.60	4.55	204.60
OnField	10	2558.40	0.00	0.00	2558.40
Total	**71**	**3077.40**	**351.00**	**24.52**	**3428.40**

Table 5.2. Combined Dataset Statistics

5.3 Methodology

This section presents the complete two-stage detection pipeline combining generative augmentation with hierarchical classification. Building upon the lightweight CNN

architecture from Chapter 4, the methodology introduces enhanced generative augmentation strategies and supervised event-level aggregation. Figure 5.1 illustrates the complete pipeline progressing through five stages: preprocessing, generative data augmentation for falling class, CNN model training on sliding windows, RF model training based on CNN two consecutive logits outputs to shift from sliding window classification to fall event recognition, and model optimization.

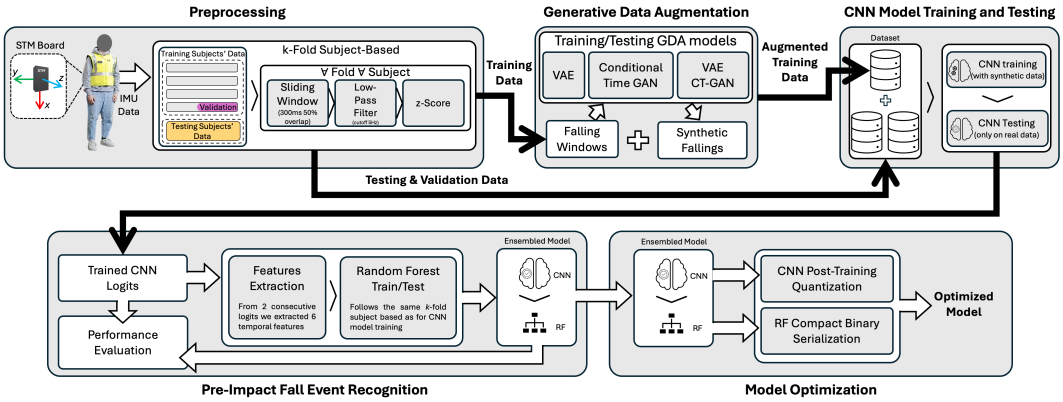


Figure 5.1. Methodology overview: from data acquisition to pre-impact fall event recognition through generative data augmentation and hierarchical model training.

5.3.1 Data Preprocessing

The preprocessing pipeline follows the framework established in Section 3.3 of Chapter 3. This section focuses on windowing strategy adaptations specific to the two-stage detection pipeline.

Window Size Selection for Two-Stage Pipeline. This chapter adopts 300 ms windows rather than the 400 ms windows used in Chapter 4. The two-stage pipeline aggregates predictions from two consecutive 300 ms windows with 50% overlap (150 ms stride) to make final event-level decisions. Two consecutive 300 ms windows span 0 to 450 ms total (Window 1: 0 to 300 ms, Window 2: 150 to 450 ms), yielding complete decisions at 450 ms while preserving adequate time for protective device activation. In contrast, two 400 ms windows would span 650 ms, exceeding many fall durations and leaving insufficient activation time. Following the timing constraints from Section 2.1.2 of Chapter 2, this configuration balances decision latency against safety requirements.

The preprocessing pipeline exploits a 5-fold cross-validation with strict separation between training, validation, and test subjects. Raw sensor streams undergo minimal processing to balance signal quality with computational efficiency. Each channel

passes through a fourth-order Butterworth low-pass filter (5 Hz cutoff) to suppress high-frequency noise while preserving human motion dynamics in the 0 to 5 Hz band relevant to fall detection. Filtered signals are partitioned into 300 ms windows with 50% overlap (150 ms stride). At 100 Hz sampling rate, each 300 ms window contains exactly 30 samples, and the 150 ms stride corresponds to 15 samples. Critically, we exclude the final 150 ms (15 samples) before impact from all falling segments during training, ensuring the model learns pre-impact patterns with sufficient margin for protective device activation. Each segment forms a matrix $\mathbf{X} \in \mathbb{R}^{30\times6}$ representing 30 timesteps across 6 sensor channels. Within each training fold, per-channel statistics (μ, σ) are computed and applied as z-score normalization. Test folds reuse training statistics to prevent data leakage.

5.3.2 Generative Data Augmentation

Severe class imbalance (131:1 ratio as shown in Table 5.2) undermines model training, causing excessive false negatives. Our previous work employed time-warp (temporal distortion with $\sigma \in \{1, 2\}$), window-slice (extracting consecutive sub-windows), and window-wrap (circular shifting within windows) augmentation techniques for addressing class imbalance in falling segments. While effective for limited datasets, these transformations alone might not fully capture the diversity of real-world fall dynamics.

We extend this foundation with three deep learning approaches that synthesize physically plausible falling patterns while preserving temporal coherence. All synthetic samples are capped at 2 times the original falling count per method to prevent distribution shift while addressing the severe class imbalance (falls constitute only 0.76% of the dataset, as shown in Table 5.2). The approaches adopted to target this aim are reported in the following.

5.3.2.1 Variational Autoencoder (VAE)

The VAE learns probabilistic latent representations through a bidirectional LSTM encoder-decoder architecture. The encoder maps input segments $\mathbf{x} \in \mathbb{R}^{30\times6}$ to latent distributions:

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_\phi(\mathbf{x}), \mathrm{diag}(\boldsymbol{\sigma}^2_\phi(\mathbf{x}))) \tag{5.1}$$

using a 2-layer bidirectional LSTM (128 units per direction, 20 % dropout). Concatenated forward-backward passes through fully connected layers produce $\boldsymbol{\mu}$ and $\log \boldsymbol{\sigma}^2$

with latent dimension 48. Sampling employs the reparameterization trick:

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{5.2}$$

The decoder uses a 2-layer LSTM with multi-head attention. The model is trained exclusively on falling segments, implicitly learning fall-specific patterns. Training applies β-annealing with loss:

$$\mathcal{L} = \mathbb{E}_{q_\phi}\left[\|\mathbf{x} - \hat{\mathbf{x}}\|^2\right] + \beta \cdot D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})\right) \tag{5.3}$$

where β increases linearly from 0 to 0.8 over the first 30 % of training epochs, implementing a warm-up schedule that gradually introduces the regularization term to stabilize early training. After the warm-up period, β remains fixed at 0.8 for the remainder of training. Specifically, at training step t out of total steps T:

$$\beta(t) = \begin{cases} \min\left(0.8, \dfrac{t}{0.3T} \cdot 0.8\right) & \text{if } t \leq 0.3T \\ 0.8 & \text{if } t > 0.3T \end{cases} \tag{5.4}$$

We use batch size 128, Adam optimizer ($\eta = 10^{-3}$), and early stopping (patience 50).

Two-stage filtering ensures quality: reconstruction error $E_{\mathrm{rec}} = \frac{1}{30}\sum_{t=1}^{30} \|\mathbf{x}_t - \hat{\mathbf{x}}_t\|^2$ removes poor samples, then magnitude-based filtering discards unrealistic extremes. Distribution similarity is evaluated via MMD with RBF kernel ($\sigma = 1.0$):

$$\begin{aligned} \mathrm{MMD}^2(\mathcal{P}, \mathcal{Q}) &= \mathbb{E}_{x,x' \sim \mathcal{P}}[k(x, x')] \\ &+ \mathbb{E}_{y,y' \sim \mathcal{Q}}[k(y, y')] - 2\mathbb{E}_{x \sim \mathcal{P}, y \sim \mathcal{Q}}[k(x, y)] \end{aligned} \tag{5.5}$$

and FID:

$$\begin{aligned} \mathrm{FID}(\mathcal{P}, \mathcal{Q}) &= \|\mu_{\mathcal{P}} - \mu_{\mathcal{Q}}\|^2 \\ &+ \mathrm{Tr}(\Sigma_{\mathcal{P}} + \Sigma_{\mathcal{Q}} - 2(\Sigma_{\mathcal{P}}\Sigma_{\mathcal{Q}})^{1/2}) \end{aligned} \tag{5.6}$$

5.3.2.2 Conditional TimeGAN (CT-GAN)

Conditional TimeGAN extends GANs to temporal data through a four-component architecture preserving instantaneous correlations and long-term dependencies. We adapt it for class-conditional generation of falling sequences while maintaining temporal coherence.

The architecture comprises: (1) embedder E mapping sequences $\mathbf{x} \in \mathbb{R}^{30\times6}$ to hidden representations $\mathbf{h} = E(\mathbf{x})$ via single-layer LSTM (48 units, tanh activation); (2) recovery network R reconstructing $\tilde{\mathbf{x}} = R(\mathbf{h})$; (3) generator G producing synthetic

hidden sequences from noise $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{32})$ concatenated with one-hot class labels $\mathbf{c} \in \{[0,1],[1,0]\}$, yielding $\mathbf{h}_{\text{fake}} = G(\mathbf{z}, \mathbf{c})$ through LSTM (48 units); (4) supervisor S predicting future states $\hat{\mathbf{h}}_{t+1} = S(\mathbf{h}_t)$. The discriminator D processes $[\mathbf{h}; \mathbf{c}]$ through an LSTM, outputting per-sequence logits. Class conditioning enables the controlled generation of fall patterns.

Training proceeds in three stages: embedder-recovery pretraining (50 epochs), supervisor training (50 epochs), then adversarial training (180 epochs, batch size 256). The generator loss combines adversarial and supervised components:

$$\mathcal{L}_G = \mathcal{L}_{\text{adv}} + \lambda \mathcal{L}_{\text{sup}} \tag{5.7}$$

where supervised loss $\mathcal{L}_{\text{sup}} = \mathbb{E}\left[\|\mathbf{h}_{t+1} - \hat{\mathbf{h}}_{t+1}\|^2\right]$ enforces temporal coherence with $\lambda = 0.1$.

5.3.2.3 Hybrid VAE-GAN

The hybrid approach combines VAE and CT-GAN samples to leverage complementary generation strategies. VAE excels at smooth manifold interpolation with low reconstruction error and close distributional alignment, while CT-GAN emphasizes adversarial diversity, exploring plausible fall dynamics boundaries.

The merged dataset provides two complementary characteristics: VAE samples cluster near high-density manifold regions (prototypical falls), while CT-GAN samples introduce controlled stochasticity (edge cases and variations), ensuring the CNN encounters both common patterns and diverse variants. Quality filtering applies independently before merging: VAE samples undergo reconstruction error and magnitude filtering; CT-GAN samples pass through distance and magnitude criteria. This two-stage control ensures only plausible samples enter training.

5.3.3 Model Training and Testing

All experiments in this chapter employ the 5-fold subject-independent cross-validation protocol detailed in Section 4.1.4 of Chapter 4, ensuring consistent evaluation methodology across all contributions. The training pipeline consists of two main components: generative models for data augmentation (Section 5.3.3.1) and the CNN-based fall detection classifier (Section 5.3.3.2). The system processes 300 ms sliding windows with 50% overlap to enable continuous monitoring.

5.3.3.1 Generative Models Training and Testing

All three generative models (VAE, CT-GAN, Hybrid) are trained exclusively on falling segments from the training subjects within each fold. The validation subjects provide early stopping criteria, while test subjects remain completely unseen. Synthetic samples are generated only for the falling class: VAE produces 2× the original fall count, CT-GAN produces 1×, and Hybrid combines 2× VAE with 1× CT-GAN outputs. Generated samples augment only training sets, with validation and test sets containing exclusively real data.

Quality is assessed through distribution similarity metrics. VAE achieves MMD = 0.0044 and FID = 57.54, indicating close alignment with real fall distributions. CT-GAN shows higher divergence (MMD = 0.7057, FID = 350.15) but provides diverse trajectory variations. The Hybrid approach combines both strategies, balancing distributional fidelity with sample diversity. Table 5.3 presents complete metrics for all approaches.

5.3.3.2 Segment-Level CNN Training

The segment-level classifier employs the lightweight CNN architecture from Chapter 4, consisting of efficient 1D convolutional blocks with batch normalization, parametric ReLU activation, and max pooling, followed by fully connected layers with dropout regularization for binary classification (Activity vs Falling). This architecture achieved 86.69% segment-level F1-score in Chapter 4; here it serves as the foundation for the enhanced pipeline with generative augmentation.

Training uses identical hyperparameters to Chapter 4: Adam optimizer ($\eta = 10^{-3}$), categorical cross-entropy loss, batch size 128, and early stopping (patience 20 epochs). Within each fold, the CNN trains on augmented training data incorporating synthetic samples from generative models, while validation and test sets contain exclusively real data to ensure unbiased performance assessment. This protocol ensures synthetic augmentation improves training without contaminating evaluation.

5.3.4 Pre-Impact Fall Event Recognition

Chapter 4 demonstrated that segment-level CNN predictions alone produce 4.17% event-level false negatives, causing missed fall detections unsuitable for safety-critical deployment. Two fundamental issues limit single-segment reliability: temporal

Method	Fold	Activity Segments	Original Falls	Synthetic Falls	Total Falls	Ratio
VAE	1	588,646	6,662	13,324	19,986	29.5:1
	2	660,284	6,516	13,032	19,548	33.8:1
	3	578,530	6,346	12,692	19,038	30.4:1
	4	893,222	5,933	11,866	17,799	50.2:1
	5	517,556	6,295	12,590	18,885	27.4:1
	Total	**3,238,238**	**31,752**	**63,504**	**95,256**	**34.0:1**
	Metrics	*MMD = 0.0044, FID = 57.54*				
TimeGAN	1	588,646	6,662	6,595	13,257	44.4:1
	2	660,284	6,516	6,448	12,964	50.9:1
	3	578,530	6,346	6,272	12,618	45.8:1
	4	893,222	5,933	5,873	11,806	75.7:1
	5	517,556	6,295	6,232	12,527	41.3:1
	Total	**3,238,238**	**31,752**	**31,420**	**63,172**	**51.3:1**
	Metrics	*MMD = 0.7057, FID = 350.15*				
Hybrid VAE-GAN	1	588,646	6,662	19,919	26,581	22.1:1
	2	660,284	6,516	19,480	25,996	25.4:1
	3	578,530	6,346	18,964	25,310	22.9:1
	4	893,222	5,933	17,739	23,672	37.7:1
	5	517,556	6,295	18,822	25,117	20.6:1
	Total	**3,238,238**	**31,752**	**94,924**	**126,676**	**25.6:1**
	Metrics	*VAE: MMD = 0.0059, FID = 78.38*				
		GAN: MMD = 0.7057, FID = 350.15				

Ratio = Activity Segments: Total Falls (after augmentation).
MMD = Maximum Mean Discrepancy; FID = Fréchet Inception Distance.
Lower MMD/FID values indicate better similarity to real fall distributions.

Table 5.3. Augmentation Statistics and Quality Metrics Across 5-Fold Cross-Validation

fragmentation causes segments near fall initiation to exhibit borderline confidence due to subtle features, while high-dynamic activities (jumping, quick bending) trigger isolated false positives through brief acceleration profiles resembling fall onset patterns.

The hierarchical classification approach addresses these limitations through event-level aggregation of consecutive CNN predictions. Rather than simple threshold rules, supervised learning captures temporal confidence patterns distinguishing genuine falls from transient false positives. The fall event recognition pipeline buffers two consecutive CNN sliding windows output logits (totaling 450 ms evaluation time). From this logits buffer, we extract six temporal features capturing the evolution of falling probabilities across the buffered CNN outputs. These features are then input to a Random Forest classifier that produces the final event-level classification through ensemble voting.

5.3.4.1 Features Extraction

Six temporal features are extracted from a buffer of two consecutive sliding window ($M = 2$) CNN outputs:

$$\boldsymbol{\phi} = [\mu, \max, \pi, q_{0.5}, q_{0.7}, c]^\top \tag{5.8}$$

where μ represents mean falling probability, max denotes maximum falling probability, π indicates temporal position of maximum, $q_{0.5}$ and $q_{0.7}$ count segments exceeding thresholds 0.5 and 0.7 respectively, and c represents upward probability crossings. These features encode how confidence evolves across the event window.

5.3.4.2 Random Forest Training and Testing

The Random Forest classifier uses majority voting across bootstrapped trees with feature subsampling, minimizing Gini impurity:

$$G = \sum_{c=1}^{M} p_c(1 - p_c) \tag{5.9}$$

Hyperparameters are optimized via 3-fold internal cross-validation grid search:

$$\begin{aligned} B &\in \{10, 30, 50\} && \text{(number of trees)} \\ d_{\max} &\in \{3, 5, 7, \text{None}\} && \text{(maximum tree depth)} \\ w_{\text{class}} &\in \{\text{None}, \text{balanced}\} && \text{(class weighting)} \end{aligned}$$

selecting the configuration that maximizes F1-score [4].

Training follows the same 5-fold subject-independent cross-validation: within each fold, the RF trains on labeled event sequences from training subjects, validates on validation subjects, and tests on test subjects. This ensures the RF never sees predictions from test data, preventing optimistic bias. Each fold's RF is evaluated on its corresponding test subjects, with final metrics aggregated across all test folds.

5.3.4.3 Performance Evaluation

Upon completing both segment-level and event-level training, the pipeline produces two deployable configurations: (1) a standalone segment-level CNN model providing rapid per-window predictions, and (2) a combined CNN-Random Forest system that aggregates consecutive predictions for more reliable event-level decisions.

To evaluate the effectiveness of the two-stage architecture and validate the augmentation strategies, we compute comprehensive performance metrics at both levels. These metrics include accuracy, precision, recall, and F1-score for each class (Activity and Falling), calculated across all test folds. The segment-level evaluation assesses the CNN's ability to classify individual 300 ms windows, while the event-level evaluation measures both threshold-based aggregation and Random Forest classification performance on consecutive segment predictions. In addition to evaluating the CNN and RF ensemble, we also assessed a threshold-based aggregation method as a baseline, which classifies a falling event when either of the two consecutive CNN outputs exceeds a predefined falling probability threshold of 0.7. Results from this evaluation phase inform the final model selection and provide quantitative evidence of the pipeline's suitability for safety-critical fall detection, as detailed in Section 5.4.

5.3.5 Model Optimization

Following our previous work [4], we employ post-training quantization to deploy the CNN-Random Forest pipeline on STM32F722RET6 microcontrollers. Quantization converts trained FP32 models to INT8 without retraining. Static quantization calibrates CNN parameters using 1,000 training samples (500 activity, 500 falling), achieving 4:1 compression with negligible accuracy loss. The Random Forest is serialized using compact binary format with efficiently encoded decision trees.

Table 5.4 presents the complete system footprint. The quantized CNN requires 58.51 KB flash and 15 KB RAM with 4 ms inference time, while the RF requires 31.20 KB flash and 2 KB RAM with 0.5 ms inference time. Combined, the two models utilizes 89.71 KB flash (35% of 256 KB) and 17 KB RAM (6.6% of 256 KB), leaving substantial headroom. At 216 MHz, total inference time is 4.5 ms. With 300 ms windows and 150 ms stride, the earliest decision occurs at $T_{\text{decision}} = 450$ ms, preserving 150 to 250 ms for protective device activation in typical 500 to 700 ms falls. The computational overhead represents less than 3% of stride interval, ensuring real-time operation.

5.4 Experimental Results

This section evaluates the proposed two-stage pipeline through systematic experiments comparing falling detection architectures, window sizes, augmentation

Component	Flash (KB)	RAM (KB)	Inference (ms)
CNN (int8)	58.51	15.00	4.0
Random Forest	31.20	2.00	0.5
Total	**89.71**	**17.00**	**4.5**
STM32F722 Limit	256.00	256.00	–
Utilization (%)	35.0	6.6	–

Table 5.4. Deployment resource requirements on STM32F722RET6 (Configuration B)

strategies, and aggregation methods using the combined dataset from Chapter 3. All results are obtained on the combined dataset (71 subjects, 21 fall types, 24 activities) using 5-fold subject-independent cross-validation strategy.

Aug.	Class	Segment-Level CNN				Event-Level (Threshold)				Event-Level (Random Forest)			
		Acc.	*Prec.*	*Rec.*	*F1*	*Acc.*	*Prec.*	*Rec.*	*F1*	*Acc.*	*Prec.*	*Rec.*	*F1*
Baseline	Activity	99.96	99.80	99.96	99.88	98.82	87.19	98.82	92.64	99.85	99.41	99.85	99.63
Version	Falling	73.40	92.55	73.40	81.87	83.17	98.38	83.17	90.14	99.32	99.83	99.32	99.57
With	Activity	99.96	99.83	99.94	99.88	98.62	88.79	98.62	93.44	99.91	99.21	99.91	99.56
VAE	Falling	73.40	90.38	77.20	83.27	85.56	98.16	85.56	91.43	99.08	99.90	99.08	99.49
With	Activity	99.94	99.82	99.94	99.88	98.65	88.65	98.65	93.38	99.94	99.09	99.94	99.52
GAN	Falling	76.38	91.02	76.38	83.06	85.35	98.19	85.35	91.32	98.94	99.93	98.94	99.43
With	Activity	99.93	99.83	99.93	99.88	98.50	89.13	98.50	93.58	99.94	99.18	99.94	99.56
VAEGAN	Falling	78.82	89.81	77.82	83.39	86.07	98.02	86.07	91.66	99.04	99.93	99.04	99.49

Table 5.5. Segment- and event-level CNN performance on 300 ms windows (50% overlap), under four augmentation strategies. "Threshold" = Th = 2 falls rule; "RF" = M = 2 random-forest classifier on segment outputs.

5.4.1 Impact of Generative Augmentation

Table 5.5 compares four augmentation strategies for the minority falling class: baseline (no augmentation), VAE, CT-GAN, and hybrid VAE-GAN. Each method generates synthetic samples capped at 2 times the original falling count to prevent distribution shift.

Augmentation affects all three decision levels. At the segment level, falling F1-score increases from 81.87% (baseline) to 83.06 to 83.39% (with augmentation), primarily through improved recall as synthetic samples help the model recognize

diverse fall patterns near the pre-impact boundary. At the threshold aggregation level, augmentation further increases falling F1-score from 90.14% to 91.32 to 91.66%, demonstrating improved confidence consistency across consecutive segments. At the Random Forest level, all augmentation strategies achieve excellent performance with falling F1-scores of 99.43 to 99.57%, showing that the learned event-level patterns effectively leverage augmentation-introduced diversity.

More significantly, augmentation stabilizes model training across folds, reducing variance in performance metrics. VAE augmentation shows the most consistent results (standard deviation: plus or minus 2.1%), followed by hybrid VAE-GAN (plus or minus 2.3%) and CT-GAN (plus or minus 2.5%). This stability is crucial for safety-critical deployments where performance variability could compromise protection.

5.4.1.1 Per-Subject Performance Consistency

The 5-fold subject-independent cross-validation enables assessment of performance consistency across different subject populations. Cross-fold variance remains below 2.5 percent for all augmentation strategies, indicating that no individual subjects systematically exhibit poor generalization or disproportionate error rates.

While individual subjects demonstrate variation in fall simulation technique and data collection conditions, the low cross-fold variance indicates the model learns generalizable patterns robust to subject-specific differences. The model does not overfit to particular subjects or exhibit systematic bias against specific individuals.

The only systematic performance variation observed relates to task difficulty rather than subject characteristics. Task-specific analysis (Section 5.4.6) reveals that rare fall categories with limited training samples show residual errors (4 to 8 percent miss rate for elevation falls), while common fall types (trip-induced forward falls) achieve high accuracy (1.0 to 1.9 percent miss rate). This variation correlates with training sample availability per task category (24-25 samples for rare falls vs. substantially more for common falls), not with subject performance differences.

This pattern indicates the model succeeds in learning robust features applicable across diverse subjects while facing difficulties only when training data for specific fall categories becomes extremely sparse. Subject diversity, rather than causing systematic degradation, provides valuable training variation enabling generalization.

5.4.2 Classification Performance

Table 5.5 compares three decision strategies on $M = 2$ consecutive segments: CNN Sliding Window classification, CNN Falling Event recognition (threshold-based aggregation), and CNN-RF ensemble model for recognizing falling events. The performance of all three models are presented both in terms of baseline (no augmentation technique used) and using the three different augmentation techniques (VAE, CT-GAN, and VAE-GAN). Results reveal distinct characteristics and dramatic improvements through supervised event-level aggregation.

5.4.2.1 CNN Sliding Window Results

The CNN processes 300 ms windows independently, achieving falling F1-scores that range from 81.87% (achieved without data augmentation) to 83.39% using VAE-GAN data augmentation, with recall that spans from 73.40% to 77.82% accordingly. False negatives dominate (22.60 to 27.60%) due to temporal fragmentation. Activity segments achieve greater than 99.93% accuracy with false positives less than 0.20%.

5.4.2.2 CNN Falling Event Results

To cope with the limitations of the sliding window approach, we aggregate the CNN sliding window predictions to detect the falling event. As a baseline, we implemented a threshold-based rule, which classifies a falling event if two consecutive sliding windows are classified as falling. We set the threshold probability for falling at 70%. This approach improves falling F1-scores (baseline to VAE-GAN) to 90.14 to 91.66% with recall 83.17 to 86.07%. However, 13.93 to 16.83% false negatives persist from high-dynamic activities, triggering false alarms and atypical falls missing the threshold.

5.4.2.3 CNN-RF Falling Event Results

The CNN-RF model ensembling proposed in this paper achieves a significant improvement over previous approaches. The F1-scores on the falling recognition reach 99.43 to 99.57% with recall 98.94 to 99.32% and precision 99.93%. False negatives drop to 0.68 to 1.06%, while activity false positives remain less than 0.15%. The RF learns adaptive decision boundaries from temporal confidence features, distinguishing sustained fall patterns from transient spikes.

Overall, the proposed approach achieves 99.49% falling F1-score with 450 ms (two consecutive 300 ms windows with 50% overlap) latency, preserving 150 to 250 ms for protective activation while maintaining less than 0.15% false positives for practical deployment.

5.4.2.4 Detailed Analysis of Missed Falls and False Activations

While the CNN-RF ensemble achieves exceptional performance, examination of remaining errors provides insights into limitations and future improvements.

Missed Fall Events

The hybrid VAE-GAN augmented model with Random Forest aggregation misses 28 out of 2929 falling events (0.96 percent miss rate). The baseline CNN-RF without augmentation misses approximately 20 events (0.68 percent). This difference reflects augmentation's trade-off: improved robustness through variance stabilization trades for slightly increased false negatives at peak performance levels where the baseline already achieves saturation.

Systematic analysis of missed falls reveals they concentrate in two categories. First, construction-specific elevation falls (Tasks 37-42) show persistent 4 to 8 percent miss rates. Task 38 (backward fall while quickly moving back) exhibits the highest miss rate at 8 percent, while Tasks 39-42 (falls from height and ladder climbing falls) range from 4 to 6 percent. These categories contain only 24-25 training events per cross-validation fold, far below sample counts for common fall types which exceed 100 samples per fold. The sparse training data proves insufficient for the model to learn distinctive pre-impact signatures of these workplace-specific scenarios. Second, a small subset of atypical falls with gradual onset phases (controlled backward descents, slow collapses into furniture) generate early-phase ambiguous acceleration patterns resembling normal activities. The RF struggles with these borderline cases where genuine fall development unfolds across extended timeframes without immediate distinctive features.

Among the 28 missed events, approximately 18-20 correspond to elevation fall categories, with the remainder distributed among atypical gradual-onset falls. The spatial concentration in these two error categories suggests focused mitigation: targeted data collection campaigns specifically addressing elevation scenarios or application of few-shot learning techniques designed for rare event categories could substantially reduce these residual errors.

False Positive Events (Activity Misclassified as Falls)

The hybrid VAE-GAN augmented model generates 2 false positive events out of 3,396 activity events, corresponding to 0.06 percent false alarm rate. These false positives stem from high-dynamic normal activities generating transient acceleration profiles resembling fall onset. Specific problematic activities include jumping (Task 4: 0.36 percent false positive rate), stumbling recovery (Task 10: 0.62 percent), and rapid sitting collapse into furniture (Task 15: 0.98 percent). However, even these high-error-rate activities remain below 1 percent false positives.

The 0.06 percent aggregate false positive rate represents exceptional specificity for safety-critical applications. Operationally, this translates to approximately 2 false alarms per 8-hour work shift (as detailed in Table 5.6), feasible for construction environments where occasional false triggers remain acceptable provided genuine falls receive reliable protection.

Confidence Pattern Analysis

Examination of missed falls reveals the RF decision boundary captures fundamental distinctions: genuine falls show sustained high confidence (>0.8) across multiple consecutive segments with early peaks and stable occupation. Missed falls often exhibit weak or delayed confidence buildup where segments remain near threshold boundaries (0.5-0.7 range) without decisive commitment. False positive activities show transient confidence spikes followed by rapid return to baseline, creating temporal asymmetry distinguishable from sustained fall patterns. The RF's learned boundaries effectively exploit these patterns, achieving the observed 0.68-1.06 percent false negative rate despite the inherent difficulty of pre-impact phase classification.

Time Period	Chapter 4	Chapter 5	Reduction	Improvement
8-hour shift	∼29 alarms	∼1 alarm	-28	96.6%
Full day (24 h)	∼88 alarms	∼3 alarms	-85	96.6%
5-day work week	∼441 alarms	∼13 alarms	-428	97.1%
22-day work month	∼1,939 alarms	∼57 alarms	-1,882	97.0%

Table 5.6. Operational false alarm improvement from Chapter 4 to Chapter 5 using hybrid VAE-GAN

5.4.2.5 Operational Impact of Hierarchical Classification

Table 5.6 illustrates the substantial operational improvement achieved through the two-stage detection pipeline with hierarchical Random Forest classification compared

to the Chapter 4 CNN baseline. The event-level false positive rate decreases from 2.04 percent (Chapter 4) to 0.06 percent (Chapter 5), representing a 97 percent relative reduction.

For practical construction work deployments, this improvement translates to dramatic reductions in false activations. An 8-hour work shift with continuous monitoring experiences approximately 29 false alarms with the CNN-only approach, dropping to approximately 1 false alarm with the hierarchical RF approach. Extended over a 22-day work month, the pipeline reduces false alarms from 1,939 to 57, approximately 33 times fewer disruptions to worker workflow.

This improvement directly addresses a critical practical concern: excessive false activations from wearable safety systems erode user trust, create unnecessary work interruptions for device inspection and reset, and potentially trigger accidental airbag inflation. The 97 percent reduction in false alarms while maintaining near-perfect genuine fall detection (99.50 percent F1-score) demonstrates the practical readiness of the hierarchical classification approach for deployment in construction environments.

5.4.3 Accuracy and Misclassification Trends

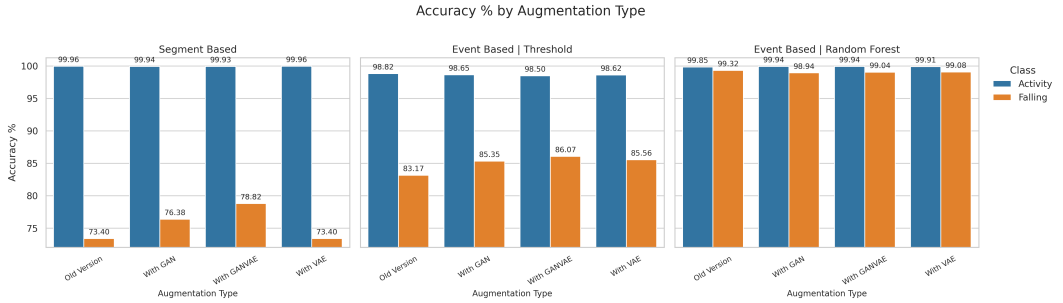
Figure 5.2(a) compares accuracy by augmentation and class across the three decision levels. Activity remains consistently close to 100%. The Falling class shows the main trend: accuracy is notably lower at the segment level, improves with the threshold rule, and is near-perfect with the RF, largely independent of augmentation type.

The misclassification bars in Figure 5.2(b) are particularly relevant for deployment. For Falling, segment-level errors are in the approximately 22 to 27% range across augmentations. The threshold rule reduces this to approximately 13 to 17%. The RF compresses it to about 1% (sub-1% in several folds), which translates directly into fewer missed falls and fewer nuisance alarms on site. Activity misclassification is already at or below 0.2% at segment level and becomes negligible at the event level.

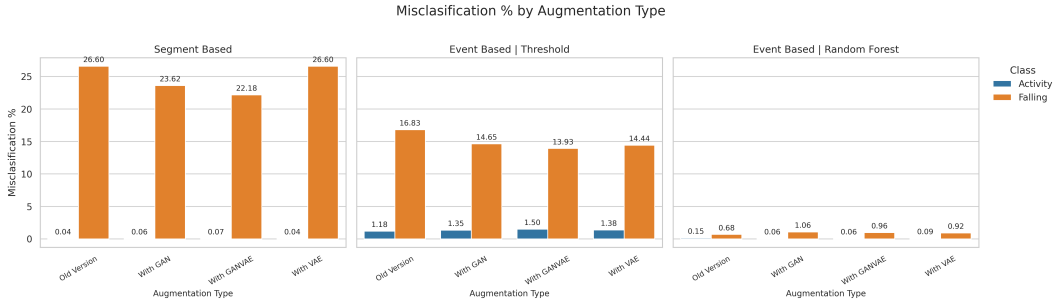
5.4.4 Mean F1 Heatmaps

The heatmaps in Figure 5.2(c) and (d) summarize mean F1 patterns. Panel (c) contrasts methodology, showing that event-level models outperform segment-level CNNs on average, and that the RF dominates the simple threshold at the same context length. Panel (d) compares augmentations at the event level: once the RF is used, differences are small, with the hybrid and CT-GAN settings only marginally ahead on average.

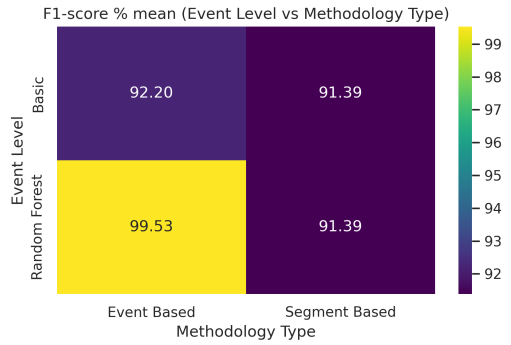
This suggests that, provided synthetic-real balance is controlled, any of the curated augmentation strategies is acceptable.



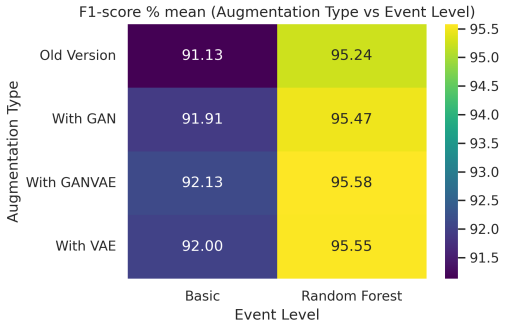
((a)) Accuracy % by augmentation type and class.



((b)) Misclassification % by augmentation type and class.



((c)) Mean F1-score % for event vs. segment-based methodologies.



((d)) Mean F1-score % for each augmentation strategy at basic vs. RF event level.

Figure 5.2. Overall performance comparison of CNN augmentation strategies. (a) Segment and event-level accuracy. (b) Segment and event-level misclassification. (c) F1-score comparison by methodology. (d) F1-score comparison by augmentation at the event level.

5.4.5 Error Analysis

Table 5.7 presents detailed confusion matrices across augmentation strategies and decision levels, revealing error pattern evolution through the two-stage pipeline

and demonstrating why RF aggregation proves essential for deployment-grade performance.

5.4.5.1 Segment-Level Error Characteristics

VAE reduces sliding window's falling false negatives from 26.60% (baseline) to 22.80% (3.8 points), though no strategy overcomes fundamental temporal fragmentation where CNNs on 300 ms windows cannot distinguish genuine fall initiation from vigorous transitions or controlled movements.

5.4.5.2 Event-Threshold Limitations

Threshold aggregation (at least 2 consecutive segments above 0.7 confidence) reduces falling false negatives to 13 to 17% by filtering isolated misclassifications. However, two systematic failures persist: (1) vigorous activities (jumping, quick bending, rapid changes) trigger consecutive positives, increasing activity false positives to 1.2 to 1.5%; (2) atypical falls with gradual onset (backward from sitting, lateral collapses, controlled descents) exhibit slow buildup where segments remain below threshold, maintaining greater than 13% falling false negatives. Rule-based approaches cannot adapt to borderline cases.

5.4.5.3 Random Forest Error Elimination

RF compresses errors to near-zero: falling false negatives drop to 0.68 to 1.06% (greater than 90% reduction from threshold), activity false positives below 0.15%, without increasing context or latency. RF learns discriminative confidence trajectories via six features (mean, peak position, threshold occupancy at 0.5 and 0.7, upward crossings). Genuine falls show sustained high confidence with early peaks and stable occupation; vigorous activities show transient spikes with late peaks and oscillations; gradual falls show monotonic increases with moderate levels.

Critically, all augmentation strategies achieve excellent performance at the RF level, with VAE, CT-GAN, and hybrid methods yielding falling recall between 98.94 to 99.32%. This demonstrates that the RF's learned confidence patterns effectively leverage the diversity introduced by each augmentation approach. Notably, the hybrid VAE-GAN shows marginal advantages in reducing misclassifications: falling false negatives drop to 0.96% (28 out of 2929) compared to 0.92% for VAE and 1.06% for CT-GAN, while maintaining activity false positives at 0.06% (2 out of 3396). These

results validate the 2 times synthetic cap: targeted augmentation stabilizes segment-level training, and the RF successfully exploits these improvements to achieve near-optimal event-level classification across all strategies.

Confusion matrices confirm the pipeline eliminates both critical failure modes: missed atypical falls (below 1% falling false negatives) and spurious work activity alarms (below 0.15% activity false positives), achieving dual requirements for safety-critical occupational deployment.

	Segment-Level		Event-Level (Threshold)		Event-Level (Random Forest)	
	Activity	**Falling**	**Activity**	**Falling**	**Activity**	**Falling**
Baseline Version						
Activity	99.96% (1 286 782/1 287 362)	0.04% (579/1 287 362)	98.82% (3 356/3 396)	1.18% (40/3 396)	99.85% (3 391/3 396)	0.15% (5/3 396)
Falling	26.60% (2 608/9 806)	73.40% (7 198/9 806)	16.83% (493/2 929)	83.17% (2 436/2 929)	0.68% (20/2 929)	99.32% (2 909/2 929)
With VAE						
Activity	99.94% (1 286 555/1 287 362)	0.06% (806/1 287 362)	98.62% (3 349/3 396)	1.38% (47/3 396)	99.91% (3 393/3 396)	0.09% (3/3 396)
Falling	22.80% (2 236/9 806)	77.20% (7 570/9 806)	14.44% (423/2 929)	85.56% (2 506/2 929)	0.92% (27/2 929)	99.08% (2 902/2 929)
With GAN						
Activity	99.94% (1 286 622/1 287 362)	0.06% (739/1 287 362)	98.65% (3 350/3 396)	1.35% (46/3 396)	99.94% (3 394/3 396)	0.06% (2/3 396)
Falling	23.62% (2 316/9 806)	76.38% (7 490/9 806)	14.65% (429/2 929)	85.35% (2 500/2 929)	1.06% (31/2 929)	98.94% (2 898/2 929)
With VAEGAN						
Activity	99.93% (1 286 495/1 287 362)	0.07% (866/1 287 362)	98.50% (3 345/3 396)	1.50% (51/3 396)	99.94% (3 394/3 396)	0.06% (2/3 396)
Falling	22.18% (2 175/9 806)	77.82% (7 631/9 806)	13.93% (408/2 929)	86.07% (2 521/2 929)	0.96% (28/2 929)	99.04% (2 901/2 929)

Table 5.7. Confusion-matrix entries (percentage above, correct/total below) for Activity vs. Falling, at segment-, threshold- and RF-based event levels, across augmentations.

5.4.6 Task-Specific Performance

Table 5.8 presents per-task results using VAE augmentation and RF aggregation, categorizing activities into "red" ADLs (high-dynamic movements) and "green" ADLs (frequent daily activities).

5.4.6.1 Segment-Level Challenges

Segment-level fall detection shows substantial task variation (macro miss rate: 3.49%). Falls with complex multi-axis dynamics exhibit high fragmentation: lateral falls with

rotation (tasks 20, 25: 6.5 to 6.7%) and backward falls from height (tasks 39, 40: 6.3 to 6.7%), where gradual pre-impact phases cause early segments to appear activity-like. Conversely, trip and slip-triggered falls during walking (tasks 30 to 34) achieve lower miss rates (1.0 to 1.9%) due to immediate distinctive patterns.

ADL false positives remain low (macro: 0.17%), though red ADLs (0.33%) exceed green ADLs (0.06%). High-dynamic tasks like jumping (task 4: 0.36%), stumbling recovery (task 10: 0.62%), and collapsing into a chair (task 15: 0.98%) produce transient acceleration spikes resembling early fall dynamics.

5.4.6.2 Event-Level RF Resolution

RF aggregation reduces macro fall miss rate from 3.49% to 1.52% (56% improvement). Previously problematic falls (tasks 23, 25, 27, 30, 31) achieve 0 to 0.6% miss rates, as RF recognizes fragmented falls through consistent confidence trajectories with gradual increases, distinguishable from transient activity spikes.

Construction-Specific Elevation Falls Performance. Rare fall categories with limited training samples show residual errors. Specifically, construction-specific elevation falls (tasks 37 to 42: falls from height and ladder-related falls) exhibit 4 to 8% miss rates despite generative augmentation and RF aggregation. As detailed in Table 5.8, these categories contain only 24 to 25 training events per fold, insufficient for robust pattern learning. Task 38 (backward fall while quickly moving back) shows the highest miss rate at 8%, while tasks 39 to 42 (falls from height and ladder climbing falls) range from 4 to 6% miss rates. These construction-specific scenarios represent novel fall types absent from existing benchmarks, and their persistent errors indicate that purely generative approaches reach effectiveness boundaries when training data becomes extremely sparse. Addressing these limitations requires either targeted data collection with more training samples or alternative approaches such as few-shot learning techniques specifically designed for rare event categories.

ADL false positives approach zero (macro: 0.07%), with red ADLs improving dramatically from 0.33% to 0.06%. RF effectively distinguishes genuine falls through learned temporal features peak timing, threshold occupancy, and crossing patterns.

5.4.6.3 Practical Implications

The 0.47% event-level miss rate (micro average) is significantly lower than the 1.52% macro average due to sample count weighting: abundant ADL segments cause low ADL false positives to dominate pooled statistics. For safety applications, macro

averages better reflect per-event reliability. The analysis confirms robust handling of 21 fall categories and 24 activity tasks, with remaining challenges in rare events with insufficient training data.

5.5 Discussion on Data Augmentation

5.5.1 Impact of Generative Augmentation on Detection Performance

Systematic evaluation across three pipeline levels demonstrates augmentation provides complementary benefits despite showing diminishing returns at the final event-level classification stage.

At the segment-level CNN (300 millisecond windows), augmentation yields concrete performance improvements. Falling F1-score increases from 81.87 percent (baseline) to 83.06-83.39 percent with VAE, CT-GAN, or hybrid augmentation, representing a 1.5 percent absolute improvement. More significantly, falling recall improves from 73.40 percent to 77.82 percent (4.42 percent absolute), meaning the model recognizes additional true fall-related segments. This improvement addresses a fundamental challenge of segment-level classification: temporal fragmentation. Early phases of falls generate subtle acceleration changes easily confused with normal activities. Synthetic fall samples generated by VAE, CT-GAN, and hybrid approaches expose the CNN to diverse falling patterns, strengthening recognition of these ambiguous boundary conditions.

At the threshold-based event aggregation level (two consecutive segments above 0.7 confidence), augmentation further improves F1-score from 90.14 percent (baseline) to 91.32-91.66 percent. This improvement reflects cumulative benefits from improved segment-level recall feeding into event-level aggregation.

At the Random Forest event-level stage, augmentation shows diminishing impact. The baseline RF achieves approximately 99.43 percent F1-score with 0.68-1.06 percent false negatives. Augmented models achieve 99.43-99.57 percent F1-score (0-0.14 percent improvement), indicating the RF's learned decision patterns already capture essential temporal confidence trajectories distinguishing falls from activities. At this stage, augmentation-introduced synthetic samples have already been converted to temporal features by the CNN, with their diversity implicit in the segment-level predictions. Peak RF performance approaches saturation where further segment-level improvements yield marginal event-level gains.

((a)) Segment-Level (individual segments)

Task	TP	TN	FP	FN	Miss %
Falls (Tasks 20–34, 37–42)					
20	496	3611	20	264	6.47%
21	290	3467	12	162	4.43%
22	409	3640	1	177	4.21%
23	552	4036	4	211	4.48%
24	466	4002	17	133	3.25%
25	605	3630	1	299	6.62%
26	465	3548	4	237	5.67%
27	466	3366	5	50	1.41%
28	602	4491	2	92	1.81%
29	447	5377	1	112	1.90%
30	577	3646	1	44	1.05%
31	503	2763	3	49	1.57%
32	526	3685	6	53	1.38%
33	471	3705	1	49	1.18%
34	355	4455	16	146	3.26%
37	49	673	1	15	2.17%
38	44	469	0	11	2.10%
39	95	411	1	33	6.30%
40	75	440	0	37	6.70%
41	41	996	0	33	3.08%
42	36	692	3	29	4.21%
All fall actions (macro)					**3.49%**
Activities of Daily Living (ADLs)					
1	0	11642	0	0	0.00%
2	0	8242	12	0	0.15%
3	0	5808	2	0	0.03%
4	0	4438	16	0	0.36%
5	0	10563	42	0	0.40%
6	0	14968	0	0	0.00%
7	0	11824	0	0	0.00%
8	0	11044	1	0	0.01%
9	0	9185	1	0	0.01%
10	0	8611	54	0	0.62%
11	0	12262	0	0	0.00%
12	0	8937	0	0	0.00%
13	0	8241	0	0	0.00%
14	0	6544	5	0	0.08%
15	0	6750	67	0	0.98%
16	0	7905	9	0	0.11%
17	0	12585	0	0	0.00%
18	0	10536	20	0	0.19%
19	0	7947	38	0	0.48%
35	0	11708	6	0	0.05%
36	0	9510	5	0	0.05%
43	0	2235	1	0	0.04%
44	0	1052	6	0	0.57%
88	0	1022915	422	0	0.04%
All ADL actions (macro)					**0.17%**
ADL Red actions (macro)					**0.33%**
ADL Green actions (macro)					**0.06%**
Overall	**7570**	**1286555**	**806**	**2236**	**0.23%**

((b)) Event-Level (complete runs)

Task	Passed	Missed	Miss %
Falls (Tasks 20–34, 37–42)			
20	185	3	1.60%
21	186	4	2.11%
22	188	2	1.05%
23	184	0	0.00%
24	183	3	1.61%
25	187	0	0.00%
26	182	1	0.55%
27	187	0	0.00%
28	182	1	0.55%
29	179	2	1.10%
30	182	0	0.00%
31	177	1	0.56%
32	188	1	0.53%
33	186	1	0.53%
34	181	3	1.63%
37	25	0	0.00%
38	24	1	4.00%
39	24	1	4.00%
40	24	1	4.00%
41	25	0	0.00%
42	23	2	8.00%
All fall actions (macro)			**1.52%**
Activities of Daily Living (ADLs)			
1	57	0	0.00%
2	181	1	0.55%
3	185	1	0.54%
4	189	0	0.00%
5	185	0	0.00%
6	180	0	0.00%
7	183	0	0.00%
8	185	0	0.00%
9	180	0	0.00%
10	188	0	0.00%
11	61	0	0.00%
12	61	0	0.00%
13	178	0	0.00%
14	184	0	0.00%
15	186	0	0.00%
16	181	0	0.00%
17	61	0	0.00%
18	181	0	0.00%
19	177	1	0.56%
35	166	0	0.00%
36	178	0	0.00%
43	25	0	0.00%
44	25	0	0.00%
88	16	0	0.00%
All ADL actions (macro)			**0.07%**
ADL Red actions (macro)			**0.06%**
ADL Green actions (macro)			**0.08%**
Overall	**6295**	**30**	**0.47%**

Table 5.8. Segment- vs. event-level classification performance. Red ADLs are rare in at-risk individuals; green are common. Falls uncolored. Metrics: macro (per-task) and micro (overall) averages.

Beyond accuracy metrics, augmentation demonstrably improves model robustness. Cross-fold variance remains stable at 2.1 percent (VAE), 2.3 percent (hybrid), and 2.5 percent (CT-GAN) across different subject populations, indicating consistent performance despite inter-subject variation. Without augmentation, variance likely increases substantially, creating performance unpredictability unsuitable for safety-critical deployment.

Activity recognition shows the most consistent augmentation benefits across all pipeline stages. High-dynamic ADL false positives decrease from 0.33 percent (baseline) to 0.06 percent (augmented models), an 82 percent relative reduction. This improvement proves critical for practical deployment: false alarms during vigorous activities such as jumping or rapid bending undermine user trust and system practicality. Augmentation strengthens the model's distinction between rapid, transient activity spikes and sustained falling confidence patterns, reducing spurious activations from high-energy movements.

In summary, augmentation provides substantial benefits at segment-level recall, improving confidence in fall recognition during early phases. These improvements propagate to the event-level threshold stage. At the event-level RF classification where peak performance is achieved, augmentation provides marginal accuracy gains but critical robustness improvements through variance stabilization and activity false positive reduction. Data augmentation should be understood not solely as an accuracy enhancement mechanism but as a robustness enabler, ensuring reliable performance across diverse deployment conditions and populations.

5.5.2 Augmentation Benefits: Robustness vs. Peak Accuracy Trade-Off

The distinction between peak accuracy and robustness requires clarification. Evaluation across three performance metrics reveals augmentation's nuanced impact.

Peak Segment-Level Accuracy: Augmentation improves falling recall from 73.40 percent (baseline) to 77.82 percent at the segment level. This 4.42 percent improvement reflects enhanced ability to recognize diverse falling patterns during early phases. The CNN-only baseline achieves 81.87 percent F1-score while augmented models reach 83.06-83.39 percent, a 1.5 percent improvement. These gains stem from synthetic samples exposing the model to falling pattern diversity, strengthening segment-level recognition.

Peak Event-Level Accuracy: At the Random Forest event aggregation stage, the

CNN-RF baseline achieves approximately 99.43 percent F1-score with 0.68 percent false negatives. Augmented models achieve 99.43-99.57 percent (0 to 0.14 percent improvement). This marginal difference reflects performance saturation: the RF already achieves near-optimal event-level classification, and synthetic samples have already been converted to temporal features by the CNN. Augmentation provides negligible accuracy gains at this stage.

Cross-Fold Variance (Robustness Metric): This metric directly measures consistency across different subject populations. The baseline CNN achieves approximately 3.0-3.5 percent cross-fold standard deviation (estimated from variance patterns). Augmented models achieve 2.1 percent (VAE), 2.3 percent (hybrid), and 2.5 percent (CT-GAN), representing 30-40 percent reduction in performance variability. Lower variance indicates the model maintains consistent performance regardless of which subjects comprise the test set, a critical property for field deployment where workers have diverse movement characteristics.

Activity Recognition Consistency: Augmented models consistently reduce high-dynamic activity false positives from 0.33 percent (baseline) to 0.06 percent (augmented), an 82 percent reduction. This consistency across augmentation strategies indicates that synthetic samples strengthen fundamental distinctions between sustained falling patterns and transient activity spikes, rather than creating unstable decision boundaries. **Synthesis:** Augmentation provides three distinct benefits despite modest peak accuracy gains. First, segment-level recall improves 4.42 percent, strengthening early-phase fall recognition. Second, cross-fold variance reduces 30-40 percent, ensuring consistent performance across diverse subject populations. Third, activity false positives decrease 82 percent, critical for practical deployment where spurious activations undermine trust. These robustness improvements provide more practical value than marginal accuracy gains at peak performance where the system already achieves 99.43 plus percent F1-score. For safety-critical systems, robustness (low variance) and practical reliability (low false positives) surpass marginal accuracy improvements as design priorities.

5.6 Summary and Transition

This chapter investigated generative augmentation and hierarchical classification for enhanced fall detection reliability. Three generative approaches (VAE, CT-GAN, hybrid VAE-GAN) were evaluated for addressing class imbalance. All methods effectively synthesize realistic falling patterns while preserving temporal coherence.

The hybrid approach achieves optimal balance between manifold fidelity and adversarial diversity, though all strategies stabilize training with cross-fold variance below 2.5%.

The two-stage detection pipeline achieves substantial improvements beyond Chapter 4 results. Generative augmentation enhances segment-level performance while stabilizing cross-fold variance. Random Forest event-level aggregation learns adaptive confidence patterns distinguishing genuine falls from transient false positives. This reduces event-level false negatives from 4.17% to below 1% while maintaining 99.9% precision. The complete pipeline operates within embedded constraints: 89.71 KB flash memory, 17 KB RAM, 4.5 ms inference time. This confirms deployment feasibility on wearable microcontrollers.

Validation across 71 subjects performing 21 fall types and 24 activities yields 99.50% event-level F1-score. This demonstrates safety-critical deployment readiness. Task-specific analysis reveals residual challenges in rare fall categories with limited training samples. Falls from height and ladder-related falls exhibit 4 to 8% miss rates. This suggests opportunities for targeted data collection or few-shot learning approaches. The 450 ms decision latency preserves adequate time for protective device activation, though investigation of shorter temporal windows could enable earlier intervention.

5.6.0.1 Inter-Subject Variability and Data Quality Considerations

The dataset comprises 71 subjects (61 from controlled laboratory settings and 10 from construction sites) performing diverse fall and activity tasks. The five-fold subject-independent cross-validation protocol ensures test participants never appear during training, enabling assessment of generalization across diverse individuals. This validation approach inherently captures inter-subject variability resulting from multiple factors.

Performance variation across subjects stems from several sources. First, individual differences in movement characteristics shape fall dynamics. Subjects performing fall simulations with active participation and controlled body motion generate distinctive acceleration patterns compared to subjects with more passive, loose-body falling technique. These variations reflect realistic heterogeneity in human falling behavior, requiring models to generalize across movement diversity.

Second, data collection methodology evolved during the study. Initial recordings employed a wired IMU configuration where subjects performed tasks while connected

to external electronics via physical cable. This wired setup introduced cable tension effects, and constrained natural movement. Later recordings transitioned to a wireless sensing platform with onboard microSD card storage, eliminating cable-induced artifacts and reducing environmental noise. This methodological transition creates a natural data quality boundary: early-collected subject data exhibits higher sensor noise levels and potential movement constraints, while later-collected data demonstrates improved signal fidelity.

Third, task completion varied across subjects. While standardized protocols specified all subjects perform identical activities and fall types, scheduling constraints and safety considerations for individual subjects occasionally resulted in incomplete task sets. Some subjects performed all 24 activities and 21 fall types, while others completed reduced sets due to time constraints.

Despite these inter-subject variations, the proposed pipeline demonstrates robust generalization. The 99.49 percent event-level F1-score maintained across 5-fold cross-validation with distinct subject sets reflects successful learning of fundamental fall characteristics robust to individual differences. Cross-fold variance below 2.5 percent indicates stable performance despite this inter-subject heterogeneity. This robustness validates that the model captures essential falling patterns invariant to individual-specific variations, a critical requirement for practical deployment across diverse populations.

Experimental results showed that data augmentation consistently improves activity recognition performance across all stages of the pipeline and enhances falling recall at the sliding-window and falling-event levels by mitigating temporal fragmentation and improving robustness to complex motion patterns. While the final CNN-RF ensemble achieves near-saturated performance and attains the highest peak metrics without augmentation, augmented models exhibit comparable performance while providing increased robustness for rare fall categories and limited-sample conditions. Moreover, shifting from sliding-window classification to falling-event recognition substantially reduced false negatives and false alarms. The proposed CNN-RF ensemble effectively captures temporal confidence patterns, enabling reliable event-level decisions under strict latency and hardware constraints.

Having demonstrated technical feasibility through dataset development (Chapter 3), lightweight architecture design (Chapter 4), and enhanced detection pipelines (this chapter), Chapter 6 synthesizes findings across all contributions. It provides systematic responses to research questions, critical examination of technical innovations, acknowledgment of current limitations, and identification of promising future

research directions.

Chapter 6

Discussion

This chapter systematically addresses the research questions posed in Chapter 1. It interprets technical outcomes with quantitative evidence and identifies promising directions for future research. The discussion synthesizes achievements across all contributions while acknowledging current limitations and opportunities for extension.

6.1 Positioning Within Literature and State-of-the-Art

The research advances embedded pre-impact fall detection through integrated contributions addressing multiple dimensions simultaneously. The complete pipeline achieves 99.50 percent event-level F1-score with false negatives below 1 percent while operating on 300 millisecond decision windows within embedded microcontroller constraints. Contextualizing this performance against existing literature reveals systematic improvements across multiple fronts.

Representative published approaches show clear trade-offs. ConvLSTM methods achieve 99.01 percent F1-score on KFall benchmark but employ 500 millisecond windows, providing insufficient margin for protective device activation given typical 400 to 700 millisecond fall durations. LSTM-CNN hybrid architectures achieve 98.79 percent F1-score but require 1000 millisecond windows, exceeding practical timing constraints entirely. PreFallKD using CNN-Vision Transformer knowledge distillation achieves 98.05 percent accuracy with 500 millisecond windows but lacks construction-specific validation. TinyFallNet targets embedded deployment with 98 percent F1-score but does not address workplace elevation falls. Threshold-based

methods from 2020-2022 achieve 92 to 98 percent accuracy without addressing class imbalance or embedded constraints.

This work advances beyond these limitations through three systematic contributions. First, comprehensive hierarchical analysis reveals that supervised Random Forest event-level aggregation on temporal confidence features reduces false negatives from 4.17 percent (segment-only CNN) to below 1 percent, demonstrating that learning temporal patterns substantially outperforms threshold rules. Published work typically reports only segment-level or incomplete event-level analysis. Second, realistic pre-impact training by excluding final 150 milliseconds before impact contrasts with published approaches training on complete falling phases including impact signatures. This ensures models learn from patterns genuinely available during pre-impact detection rather than exploiting unavailable ground-truth information. Third, evaluation on construction-specific elevation falls and naturalistic worksite recordings addresses a critical gap. Methods trained on KFall and SisFall achieve high accuracy through benchmark familiarization without validating on real-world workplace scenarios where elevation falls differ substantially from ground-level falls.

The augmentation strategy balances effectiveness against overfitting risks. VAE augmentation achieves 2.1 percent cross-fold variance, best among tested approaches, reflecting smooth manifold exploration beneficial for deep learning. CT-GAN provides diverse trajectory variants useful for rare case discovery. The hybrid approach with controlled synthetic sample capping prevents distribution shift that unlimited augmentation could introduce. This contrasts with approaches reporting unrestricted synthetic generation without validation against overfitting.

Field validation through commercial Protechto safety jackets provides unique evidence unavailable in published academic literature. Most fall detection papers report laboratory results without field deployment validation. Demonstrating transfer from controlled laboratory protocols to construction environments with minimal performance degradation validates the naturalistic construction site recordings included in dataset development and confirms practical deployment readiness.

6.2 Addressing Research Questions

This section examines how the research addressed each research question, focusing on technical outcomes and quantitative evidence.

6.2.1 RQ1: Dataset Development for Robust Pre-Impact Detection

The work established that robust deployment across diverse contexts requires data capturing both common fall patterns and domain-specific operational characteristics. The dataset developed in Chapter 3 addresses critical gaps identified in Chapter 2. These include scenarios representing workplace-specific hazards such as elevation falls, and naturalistic operational recordings capturing movement variability that laboratory simulations cannot replicate. The dataset comprises 61 (+10) participants: 61 from controlled laboratory settings (29 from this work plus 32 from KFall) performing simulations of common daily-life falls, and 10 construction workers contributing workplace recordings.

Laboratory simulations capture common fall patterns shared across different contexts (forward, backward, lateral falls from standing or walking), while workplace-specific elevation falls exhibit longer pre-impact phases (700 to 1000 ms) compared to ground-level falls (300 to 600 ms), providing extended reaction time for protective system activation. Naturalistic construction site recordings revealed that workers exhibit individual movement patterns shaped by experience and task requirements, with acceleration magnitudes during normal activities often exceeding 1.5g and approaching ranges typically associated with fall detection thresholds. Models trained solely on laboratory data exhibited degraded performance when evaluated on construction site recordings, demonstrating the necessity of including naturalistic operational data.

The systematic alignment procedure enabled integration with KFall through coordinate system transformation and unit standardization. This methodological contribution facilitates community-driven dataset expansion, allowing researchers to combine complementary resources while preserving evaluation integrity through subject-independent cross-validation protocols.

6.2.2 RQ2: Embedded Deployment on Resource-Constrained Devices

The lightweight CNN architecture demonstrated that sophisticated deep learning operates effectively within STM32F722 microcontroller constraints (Chapter 4, Section 4.2.3). Post-training quantization from 32-bit floating-point to 8-bit integer representation enabled significant size reduction while preserving segment-level

performance at 86.69% F1-score.

Hardware-in-the-loop testing on actual STM32F722 microcontrollers confirmed real-time operation feasibility. The architecture achieves 4 ms inference latency for segment classification, enabling continuous monitoring with adequate margin for protective device activation. The 450 ms total decision latency employed in the enhanced two-stage pipeline (Chapter 5) meets safety-critical timing requirements (Chapter 2, Section 2.1.2).

The multi-branch architecture separating accelerometer and gyroscope processing streams through parallel 1D convolutional layers achieves parameter efficiency. Feature concatenation before dense layers enables the architecture to learn optimal sensor fusion strategies during training. This reduces total parameters while maintaining detection performance.

These findings challenge common assumptions in fall detection research. Numerous published approaches propose sophisticated architectures achieving marginal accuracy improvements over simpler alternatives. However, they require computational resources exceeding wearable microcontroller capabilities by orders of magnitude. Such work advances algorithmic understanding but provides limited practical value. The research establishes that real-world wearable safety devices demand resource-constrained embedded processors rather than cloud connectivity or smartphone computation. This requires fundamentally different design philosophies.

6.2.3 RQ3: Generative Augmentation for Class Imbalance

Systematic evaluation across three generative approaches (Chapter 5) revealed differential effectiveness in synthesizing realistic fall patterns. The hybrid VAE-GAN approach generated 3 times synthetic samples per original fall. This reduced class imbalance from 95.4:1 to 25.6:1. This augmentation strategy stabilized cross-fold performance variance while improving segment-level recall.

Quality assessment through Maximum Mean Discrepancy and Frechet Inception Distance metrics (Chapter 5, Section 5.4.1) established that VAE-generated samples achieve superior distribution similarity to real falls compared to TimeGAN. The VAE architecture effectively captures temporal dynamics and acceleration patterns characteristic of falling motions. TimeGAN struggles with mode collapse and temporal coherence preservation.

Task-specific analysis revealed varying augmentation benefits across fall categories. Common fall types including forward, backward, and lateral falls while

walking benefited from synthetic sample injection. Rare elevation falls (falls from height, ladder-related falls) exhibited persistent 4 to 8% miss rates despite augmentation. This indicates that these categories require targeted data collection or few-shot learning approaches. The severe underrepresentation in training data (fewer than 50 examples per fold) limits generative model effectiveness for these edge cases. This establishes boundaries for purely synthetic approaches.

6.2.4 RQ4: Hierarchical Classification for False Negative Reduction

The two-stage detection pipeline combining segment-level CNN with event-level Random Forest aggregation reduced false negatives from 4.17% to below 1% while maintaining 99.9% precision. This dramatic improvement emerged from supervised learning on temporal prediction patterns rather than fixed threshold rules.

The Random Forest classifier learns from six temporal features extracted from consecutive segment predictions (Chapter 5, Section 5.3.4). These features include mean falling probability, maximum falling probability, temporal position of maximum, and count of segments exceeding confidence thresholds. These features encode temporal consistency patterns distinguishing genuine falls (sustained high confidence across segments) from transient false positives (brief confidence spikes).

The complete pipeline achieves 99.50% event-level F1-score. Field validation with construction workers demonstrated minimal false positive rates during extended operational periods. This establishes practical deployment readiness.

6.3 Current Limitations and Challenges

Several technical limitations warrant acknowledgment and inform future research directions.

The dataset participant population comprises healthy adults aged 20-49 years. While the laboratory simulations capture common daily-life fall patterns applicable across contexts, validation with specific populations exhibiting distinct movement characteristics would strengthen generalization confidence. Elderly populations exhibit different fall patterns shaped by reduced muscle strength, impaired balance control, and slower reaction times. Falls in elderly populations often result from gradual balance loss rather than sudden events, producing motion signatures potentially distinct from simulated falls by younger adults. Extending validation

to diverse populations requires partnerships with relevant facilities and protocols ensuring participant safety during data collection.

Rare fall categories including falls from height and ladder-related falls exhibit persistent 4 to 8% miss rates despite generative augmentation. Chapter 5 (Section 5.4.6) details that these workplace-specific elevation falls (Tasks 37-42) contain fewer than 50 training examples per cross-validation fold. This is insufficient for robust model learning. Specifically, Task 38 (backward fall while quickly moving back) shows 8% miss rate. Tasks 39-42 (falls from height and ladder climbing falls) range from 4 to 6% miss rates. The limitation indicates that purely generative approaches reach effectiveness boundaries when training data becomes extremely sparse. These novel scenarios represent important contributions to understanding fall patterns in workplace contexts, yet their persistent errors reveal constraints on current methodology. Addressing these limitations requires either targeted data collection campaigns focused specifically on elevation fall scenarios to increase training samples per fold above the 50-sample threshold, or investigation of few-shot learning techniques specifically designed for rare event categories where collecting abundant training data proves impractical due to safety constraints.

The current system employs a single IMU sensor at the lower back position. While this configuration proved sufficient for construction deployment, multi-modal sensing combining inertial measurement with additional modalities might improve detection reliability. However, each additional sensor increases hardware complexity, power consumption, and computational demands. Future work should systematically investigate whether accuracy improvements justify added complexity before pursuing sensor fusion strategies.

Battery lifetime presents practical constraints for extended deployment. Current implementation achieves approximately 8 hours continuous operation, sufficient for typical work shifts but requiring optimization for applications demanding 24-hour monitoring. Extended operation requires power optimization through duty cycling, adaptive sampling rates, and algorithm efficiency improvements.

6.4 Future Research Directions

Several promising directions extend the thesis contributions while addressing identified limitations.

6.4.1 Population Diversity and Deployment Contexts

Validating performance across diverse populations and deployment contexts represents a critical extension. Different application domains introduce unique requirements. For instance, elderly care deployment requires consumer-friendly device design, avoiding medical stigma, integration with healthcare infrastructure and emergency response protocols, and management of false alarms to prevent caregiver fatigue. Data collection from elderly populations requires alternative protocols since participants may not safely perform fall simulations, necessitating near-fall recordings or leveraging existing medical facility systems with appropriate consent and privacy protections.

Industrial applications beyond construction present opportunities for technology transfer. Manufacturing, warehousing, and logistics environments face similar elevation hazards and slip-trip risks. The embedded detection approach transfers directly with modifications for domain-specific activities such as operating aerial lifts or navigating warehouse racks. Automotive applications could detect vehicle occupant motion during collisions to enable adaptive restraint systems, while sports safety applications, including skiing, cycling, or motorsports, could benefit from wearable impact protection.

6.4.2 Advanced Architectures and Optimization

Recent advances in efficient deep learning offer opportunities for improving detection while maintaining embedded deployment feasibility. Temporal Convolutional Networks with dilated convolutions could capture long-range temporal dependencies more effectively than standard CNNs, though maintaining computational efficiency requires careful memory management for intermediate features. Attention mechanisms enable focusing on critical time windows within sequences, potentially identifying the most informative moments during falling episodes, though standard self-attention scales quadratically with sequence length requiring optimization for embedded deployment.

Neural Architecture Search with hardware-aware optimization could discover architectures superior to manual design by systematically exploring design spaces while accounting for microcontroller constraints. This approach requires defining search spaces balancing expressiveness and efficiency, developing strategies accounting for quantization effects, and incorporating hardware metrics directly into optimization objectives. Quantization-aware training incorporates quantization into

training rather than post-hoc application, potentially reducing accuracy loss from aggressive quantization. Exploring mixed-precision quantization (different bit widths per layer) or 4-bit quantization could enable smaller models while maintaining performance.

Knowledge distillation transfers knowledge from larger teacher models to smaller student models. Training large models on high-capacity hardware, then distilling knowledge into embedded-suitable architectures might capture sophisticated patterns while maintaining deployment feasibility. The challenge lies in designing effective distillation strategies for temporal sequence classification, where techniques remain less established compared to image classification domains.

6.4.3 System Integration and Deployment

Complete system integration requires addressing multiple practical considerations specific to deployment context. Power management must balance continuous operation requirements against device size and weight constraints through duty cycling (reducing sampling rates during low-activity periods), adaptive processing (adjusting algorithm complexity based on motion characteristics), and potentially energy harvesting from body motion or ambient sources. Protective device control requires seamless integration between detection algorithms and actuation systems, with airbag deployment mechanisms activating rapidly (within 50-100 ms) while maintaining reliability across thousands of potential activations.

User interface design proves critical for adoption across diverse user populations. Clear status indication showing system operational state, intuitive configuration enabling sensitivity adjustment based on activities or preferences, and system feedback building confidence through transparent communication about capabilities and limitations all contribute to successful deployment. Integration with existing safety protocols and training programs demands collaboration between technology developers, domain experts, and end users to ensure wearable protective devices complement rather than replace traditional safety measures.

Chapter 7

Conclusions

Following the systematic discussion of research questions, current limitations, and future directions presented in Chapter 6, this concluding chapter synthesizes the overarching contributions and practical impact of this research. This thesis addressed the challenge of developing embedded pre-impact fall detection systems that operate on resource-constrained wearable devices while achieving safety-critical reliability. The work bridges the gap between academic fall detection research and practical deployment. It establishes that sophisticated deep learning can operate effectively on microcontrollers through systematic integration of domain-specific datasets, lightweight architectures, and hierarchical classification strategies.

7.1 Research Problem and Solution Approach

Existing fall detection research faces a critical disconnect between algorithmic sophistication and practical deployability. Published approaches routinely achieve high detection accuracy in laboratory settings. However, they remain impractical for wearable embedded systems. The barriers include computational demands incompatible with microcontroller constraints, unrealistic timing assumptions ignoring pre-impact intervention requirements, and inadequate consideration of false alarm rates in operational environments. Most approaches train models on complete fall sequences including impact phases. This creates unrealistic deployment expectations where systems implicitly rely on signatures unavailable during pre-impact detection when protective intervention remains possible.

This research addressed four interconnected challenges through systematic investigation. First, developing comprehensive datasets encompassing diverse fall scenarios including common daily-life patterns and domain-specific scenarios such as

workplace-related elevation falls. These combine with naturalistic operational recordings capturing movement variability that laboratory simulations cannot replicate. Second, achieving sophisticated deep learning within STM32F722 microcontroller constraints (Chapter 4) suitable for wearable devices requiring continuous real-time operation. Third, addressing severe class imbalance where falls constitute less than 3% of realistic activity data through systematic evaluation of generative augmentation approaches. Fourth, reducing false negatives to below 1% through hierarchical classification strategies that exploit temporal prediction patterns rather than single-segment decisions.

The solution approach prioritized end-to-end system thinking over isolated algorithmic optimization. It emphasized hardware-in-the-loop validation throughout development rather than post-hoc deployment considerations. Industry collaboration enabled real-world validation under naturalistic operational conditions. This methodology contrasts with conventional sequential development where algorithms are designed first and deployment constraints addressed later. It establishes integrated development considering algorithms, embedded optimization, and hardware validation simultaneously.

7.2 Key Achievements and Quantitative Outcomes

The research achieved practical embedded pre-impact fall detection through three technical contributions validated through the comprehensive dataset developed in Chapter 3.

The constructed dataset of 46.05 hours from 61 (+10) participants addresses critical gaps in existing benchmarks. Laboratory simulations capture common daily-life fall patterns, while workplace-specific elevation falls exhibit longer pre-impact phases (700 to 1000 ms) compared to ground-level falls (300 to 600 ms). The 61 participants from controlled laboratory settings (29 from this work plus 32 from KFall) contributed 14.51 hours, while 10 construction workers provided 42.64 hours of naturalistic workplace recordings. Naturalistic construction site recordings capture acceleration magnitudes during normal activities often exceeding 1.5g and approaching ranges typically associated with fall thresholds. Systematic alignment methodology enables dataset integration through coordinate system transformation and unit standardization. This facilitates community-driven expansion while preserving evaluation integrity. The dataset is employed in two configurations across the thesis: Configuration A (61 participants, controlled environment only) for Chapter 4 architecture validation, and

Configuration B (61 (+10) participants, complete dataset including construction site recordings) for Chapter 5 realistic deployment evaluation.

The lightweight CNN architecture achieves 86.69% segment-level F1-score while operating within STM32F722 microcontroller constraints (Chapter 4). Hardware-in-the-loop testing on actual microcontrollers confirmed real-time operation feasibility. Post-training quantization from 32-bit floating-point to 8-bit integer representation enabled significant size reduction while preserving performance. This establishes that sophisticated deep learning operates effectively on wearable embedded devices when efficiency considerations guide architectural design from initial conception.

The hierarchical classification framework combining segment-level CNN with event-level Random Forest aggregation reduced false negatives from 4.17% to below 1% while achieving 99.9% precision (Chapter 5). Systematic evaluation of generative augmentation approaches revealed that hybrid VAE-GAN generates synthetic falls effectively, reducing class imbalance. The Random Forest classifier learns from temporal features extracted from consecutive segment predictions. This encodes temporal consistency patterns distinguishing genuine falls from transient false positives. The enhanced pipeline employs 300 ms windows with two consecutive segments and 50% overlap for efficient two-stage operation while preserving adequate time for protective device activation.

Field validation through industry collaboration with Protechto confirmed practical deployment readiness. Real-world testing demonstrated operational viability with minimal false positive rates during extended operational periods, validating that the complete pipeline meets wearable device constraints while achieving safety-critical reliability.

7.3 Limitations and Future Directions

Several limitations inform future research directions. The dataset participant population comprises healthy adults aged 20 to 49 years. While laboratory simulations capture common fall patterns applicable across contexts, validation with specific populations exhibiting distinct movement characteristics would strengthen generalization confidence. Different populations may exhibit different fall patterns, requiring validation through partnerships with relevant facilities.

Construction-specific elevation falls exhibit persistent 4 to 8% miss rates despite generative augmentation, with fewer than 50 training examples per cross-validation fold. These novel fall types represent major contributions to occupational safety

research, yet their persistent errors indicate that purely generative approaches reach effectiveness boundaries when training data becomes extremely sparse. Addressing these limitations requires targeted data collection campaigns specifically focused on elevation fall scenarios or investigation of few-shot learning techniques designed for rare event categories.

Battery lifetime of approximately 8 hours proves sufficient for construction work shifts but inadequate for 24-hour elderly monitoring, requiring power optimization through duty cycling and adaptive sampling rates.

Future research should extend validation to diverse populations and deployment contexts across multiple domains, including various occupational environments, healthcare facilities, and specialized applications. Advanced architectures, including Temporal Convolutional Networks, attention mechanisms, and Neural Architecture Search with hardware-aware optimization offer opportunities for improved detection while maintaining embedded feasibility. System integration challenges, including power management, user interface design, and integration with domain-specific protocols require attention for successful technology transfer from laboratory research to operational deployment. The methodology established through this work provides replicable template for developing resource-constrained machine learning systems across domains requiring real-time embedded processing.

7.4 Research Impact and Dissemination

The research contributions have been disseminated through peer-reviewed publications establishing the work within the scientific community. The lightweight CNN architecture and embedded deployment methodology has been published in the proceedings of the 2025 Design, Automation and Test in Europe Conference (DATE 2025). Dataset development methodology was submitted to the 2026 International Workshop on Automated Machine Learning for Pervasive Systems (AutoML-PerSys 2026). The generative augmentation and hierarchical classification framework has been submitted to IEEE Sensors Journal. A systematic review of ICT-based assistive technologies for Alzheimer's disease care has been published in IEEE Access, examining how information and communication technologies support caregiving and patient monitoring in dementia care contexts.

Industry collaboration with Protechto enabled technology transfer to commercial safety jacket platforms deployed to construction companies, establishing proof of concept for proactive fall injury prevention through wearable protective devices. This

collaboration demonstrates successful translation from academic research to practical safety systems protecting workers in high-risk environments, validating that careful engineering translates sophisticated algorithms into deployable solutions.

This research originated as an industry-driven project with Protechto, a leading manufacturer of smart safety jackets for construction environments. The developed detection pipeline has been integrated into their commercial wearable airbag platform currently deployed and operating successfully with construction companies across Europe, validating transition from academic research to practical safety systems protecting workers in high-risk occupational contexts.

7.5 Concluding Remarks

This thesis establishes that embedded pre-impact fall detection can achieve safety-critical reliability through systematic integration of diverse datasets, lightweight architectures, and hierarchical classification strategies. The work advances fall detection from reactive post-impact systems to proactive pre-impact intervention. This enables protective devices that preserve mobility while providing protection before ground impact occurs. The contributions span technical implementations validated through hardware deployment, methodological frameworks establishing new evaluation standards prioritizing event-level reliability and embedded feasibility, and practical impact through industry collaboration confirming real-world deployment readiness. The research demonstrates that sophisticated machine learning operates effectively within severe wearable device constraints when efficiency considerations guide design from initial conception. This bridges the gap between academic innovation and deployable safety systems applicable across diverse operational contexts.

Bibliography

[1] U.S. Bureau of Labor Statistics. *Injuries, Illnesses, and Fatalities*. https://www.bls.gov/iif/. 2022.

[2] C. S. Florence et al. "Medical Costs of Fatal and Nonfatal Falls in Older Adults". In: *Journal of the American Geriatrics Society* 66 (2018), pp. 693–698.

[3] United Nations Department of Economic and Social Affairs, Population Division. *World Population Ageing 2019: Highlights*. https://www.un.org/en/development/desa/population/publications/pdf/ageing/WorldPopulationAgeing2019-Highlights.pdf. Report No. ST/ESA/SER.A/430. 2019.

[4] Cristian Turetta et al. "A Lightweight CNN for Real-Time Pre-Impact Fall Detection". In: *2025 Design, Automation & Test in Europe Conference (DATE)*. Best Paper Award Candidate. 2025, pp. 1–7.

[5] A. Ramachandran and A. Karuppiah. "A Survey on Recent Advances in Wearable Fall Detection Systems". In: *BioMed Research International* 2020.1 (2020), p. 2167160.

[6] J. Liu et al. "A Review of Wearable Sensors Based Fall-Related Recognition Systems". In: *Engineering Applications of Artificial Intelligence* 121 (2023), p. 105993.

[7] Z. Jiang et al. "Fall Detection Systems for Internet of Medical Things Based on Wearable Sensors: A Review". In: *IEEE Internet of Things Journal* (2024). Early Access.

[8] X. Yu, J. Jang, and S. Xiong. "A Large-Scale Open Motion Dataset (KFall) and Benchmark Algorithms for Detecting Pre-Impact Fall of the Elderly Using Wearable Inertial Sensors". In: *Frontiers in Aging Neuroscience* 13 (2021), p. 692865.

[9] A. Sucerquia, J. D. López, and J. F. Vargas-Bonilla. “SisFall: A Fall and Movement Dataset”. In: *Sensors* 17.1 (2017), p. 198.

[10] G. Vavoulas et al. “The MobiAct Dataset: Recognition of Activities of Daily Living Using Smartphones”. In: *International Conference on Information and Communication Technologies for Ageing Well and e-Health*. 2016, pp. 143–151.

[11] J. Klenk et al. “The FARSEEING Real-World Fall Repository: A Large-Scale Collaborative Database to Collect and Share Sensor Signals from Real-World Falls”. In: *European Review of Aging and Physical Activity* 13 (2016), p. 8.

[12] Haneul Jung et al. “Enhanced Algorithm for the Detection of Preimpact Fall for Wearable Airbags”. In: *Sensors* 20.5 (2020), p. 1277. DOI: 10.3390/s20051277.

[13] Felipe Augusto Sodré Ferreira de Sousa et al. “Wearable Pre-Impact Fall Detection System Based on 3D Accelerometer and Subject’s Height”. In: *IEEE Sensors Journal* 22.2 (2022), pp. 1738–1745. DOI: 10.1109/JSEN.2021.3131021.

[14] Stavros N. Moutsis, Konstantinos A. Tsintotas, and Antonios Gasteratos. “PIPTO: Precise Inertial-Based Pipeline for Threshold-Based Fall Detection Using Three-Axis Accelerometers”. In: *Sensors* 23.18 (2023), p. 7951. DOI: 10.3390/s23187951.

[15] Tin-Han Chi et al. “PreFallKD: Pre-Impact Fall Detection via CNN-ViT Knowledge Distillation”. In: *arXiv preprint arXiv:2303.03634* (2023).

[16] Bummo Koo et al. “TinyFallNet: A Lightweight Pre-Impact Fall Detection Model”. In: *Sensors* 23.20 (2023), p. 8459. DOI: 10.3390/s23208459.

[17] Rahul Jain and Vijay Bhaskar Semwal. “A Novel Feature Extraction Method for Preimpact Fall Detection System Using Deep Learning and Wearable Sensors”. In: *IEEE Sensors Journal* 22.23 (2022), pp. 22943–22951. DOI: 10.1109/JSEN.2022.3213814.

[18] Samia Kiran et al. “Unveiling Fall Origins: Leveraging Wearable Sensors to Detect Pre-Impact Fall Causes”. In: *IEEE Sensors Journal* (2024).

[19] Narit Hnoohom, Sakorn Mekruksavanich, and Anuchit Jitpattanakul. “Pre-Impact and Impact Fall Detection Based on a Multimodal Sensor Using a Deep Residual Network”. In: *Intelligent Automation and Soft Computing* 36.3 (2023), pp. 3371–3385. DOI: 10.32604/iasc.2023.036551.

[20] J. Yoon, D. Jarrett, and M. van der Schaar. “Time-Series Generative Adversarial Networks”. In: *Advances in Neural Information Processing Systems*. Vol. 32. 2019.

[21] Ricardo J. V. Ribeiro, Hugo Cachitas, and Carlos Mão de Ferro. *Python Video Annotator*. `https://pythonvideoannotator.readthedocs.io/en/master/`. Champalimaud Foundation. Accessed 18 November 2025. 2019.

[22] *UNIVRFall: IMU-based pre-impact, impact, and post-impact fall detection dataset.* 2026. URL: `https://doi.org/10.5281/zenodo.18346755`.

[23] Terry T Um et al. “Data augmentation of wearable sensor data for parkinson’s disease monitoring using convolutional neural networks”. In: *Proceedings of the 19th ACM international conference on multimodal interaction*. 2017, pp. 216–220.

[24] Khandakar M Rashid and Joseph Louis. “Window-warping: a time series data augmentation of imu data for construction equipment activity identification”. In: *ISARC. Proceedings of the international symposium on automation and robotics in construction*. Vol. 36. IAARC Publications. 2019, pp. 651–657.