– RESEARCH PROPOSAL –

AI-POWERED AUDIO-VISUAL INTRUDER DETECTION SYSTEM FOR SMART HOMES

BY

Desmond Makhubela
Student Number: 214396874

Submitted in fulfilment of the requirements for the degree
N. Dip. Computer Systems Engineering
At the Department of Computer Systems Engineering
In the
FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
At the **TSHWANE UNIVERSITY OF TECHNOLOGY**

Supervisors:
Munguakonkwa Emmanuel Migabo
Oluwasogo Moses Olaifa
Chunling Du

Abstract

This research proposal outlines the development and implementation of an AI-powered audio-visual intruder detection system for smart home environments. Traditional security mechanisms, such as motion sensors and closed-circuit television (CCTV), are plagued by high false alarm rates, leading to reduced user trust and effectiveness. The research problem addresses the persistent limitations in conventional intruder detection systems by proposing to build a practical, cost-effective multimodal AI system that combines audio and visual processing for enhanced threat detection accuracy. The proposed methodology employs a comprehensive system development approach, including hardware platform selection (Raspberry Pi for edge computing), data collection and preprocessing, AI model development and training using convolutional neural networks (CNNs) and recurrent neural networks (RNNs), and multimodal fusion implementation. The system will integrate real-time audio analysis through microphone arrays with visual processing via camera modules, implementing advanced fusion techniques (early, late, and hybrid) to minimize false alarms while maintaining high detection accuracy. Expected outcomes include a functional prototype system achieving ¿95

Contents

Glossary

Table 1: Glossary of Terms

Acronym	Definition
AI	Artificial Intelligence
CCTV	Closed-Circuit Television
CNN	Convolutional Neural Network
IoT	Internet of Things
MFCC	Mel-Frequency Cepstral Coefficients
PIR	Passive Infrared
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
RNN	Recurrent Neural Network
SLR	Systematic Literature Review

1 Introduction

1.1 Background to the Study

Residential security remains a paramount global concern, transcending geographic and socio-economic boundaries. The threat of intrusions, burglaries, and unauthorized access not only jeopardizes property but also endangers personal safety and well-being. The technological foundation for addressing these security challenges has evolved significantly, providing opportunities to develop more sophisticated and reliable detection systems. Current implementations predominantly rely on passive infrared (PIR) motion detectors and closed-circuit television (CCTV) cameras, but these single-modality approaches exhibit significant limitations that can be addressed through multimodal AI system development.

The primary challenge in building effective home security systems lies in the high false alarm rates caused by non-threatening stimuli such as pet movements, environmental changes (e.g., shadows or wind-blown objects), or ambient noises [1]. These frequent false positives contribute to 'alarm fatigue,' where users become desensitized to alerts, potentially ignoring genuine threats and undermining system efficacy [2]. The technical challenge of differentiating actual intrusions from routine household activities due to variability in lighting, acoustics, and spatial configurations [3, 4] presents an opportunity for developing intelligent multimodal systems that can provide contextual understanding and significantly reduce false alarms.

The solution approach involves integrating multimodal sensors—combining audio (e.g., microphones capturing anomalous sounds) and visual (e.g., cameras detecting motion)—with intelligent processing to provide enhanced contextual discernment [5, 6]. Artificial intelligence (AI) and machine learning (ML) paradigms, such as convolutional neural networks (CNNs) for visual analysis and spectrogram-based audio processing, enable adaptive threat identification that surpasses traditional rule-based approaches and forms the technical foundation for the proposed system implementation.

The implementation strategy addresses the challenge of high computational costs and complex setups by utilizing cost-effective platforms like Raspberry Pi and Arduino to democratize access while optimizing performance [7, 8]. The proposed system will implement visual techniques including object detection and activity recognition using CNNs, human identification, and anomaly detection optimized for residential environments. Audio processing will involve transforming signals into spectrograms for event classification via CNNs or RNNs, building on established surveillance audio analysis techniques.

The multimodal fusion approach will synergize audio and visual modalities to enhance reliability—for example, correlating breaking glass sounds with window motion detection. The system will leverage recent AI innovations, including transformers and attention mechanisms, to model cross-modal dependencies and improve detection accuracy. This technological foundation provides the basis for developing a practical, deployable system that addresses current limitations in home security.

The implementation will focus on edge computing solutions that enable real-time processing while maintaining privacy through local data processing. The system architecture will integrate IoT capabilities for smart home integration while implementing robust security measures and user-friendly interfaces for widespread adoption.

1.2 Problem Statement

Despite rapid progress in intelligent home security research, practical implementation of AI-driven audio-visual intruder detection systems remains limited. Existing commercial systems often depend on unimodal inputs (audio or visual) or basic AI applications, perpetuating issues like false alarms and environmental vulnerabilities. There is a critical need for a practical, cost-effective multimodal AI system that can achieve high detection accuracy while minimizing false alarms in real-world residential environments.

The technical implementation challenges include integrating multiple sensor modalities effectively, optimizing AI models for resource-constrained edge computing platforms, and developing robust fusion strategies that can operate reliably in diverse home environments. Current systems lack the sophisticated multimodal processing capabilities needed to distinguish between genuine security threats and benign household activities, resulting in user dissatisfaction and system abandonment.

This research addresses these implementation challenges by developing and deploying a complete AI-powered audio-visual intruder detection system. The system will integrate advanced multimodal fusion strategies with optimized AI algorithms, targeting high performance (¿95

1.3 Motivation

This research is motivated by several critical factors that highlight the urgent need for developing and implementing practical AI-powered audio-visual intruder detection systems:

Implementation Motivation: Current home security systems suffer from significant limitations, particularly high false alarm rates that undermine user trust and system effectiveness. There is an urgent need to develop and deploy a practical multimodal AI system that can demonstrate superior performance in real-world residential environments. Building such a system will provide concrete evidence of the feasibility and benefits of AI-powered security solutions.

Practical Motivation: Homeowners require affordable, reliable security solutions that can be easily deployed and maintained. The escalating demand for effective residential security, coupled with the inadequacies of conventional systems, necessitates the development of cost-effective AI-powered alternatives. There is a critical need to build and validate systems that minimize false alarms, enhance user trust, and provide measurable security improvements.

Technological Motivation: Rapid advancements in AI, edge computing, and IoT technologies provide the foundation for developing sophisticated yet affordable security systems. The availability of cost-effective hardware platforms like Raspberry Pi, combined with advances in lightweight AI models, makes it feasible to implement complex multimodal processing at the edge. Developing such systems will demonstrate the practical application of these emerging technologies.

Societal Motivation: By developing and validating an affordable, effective security system, this research contributes to democratizing access to advanced home security technologies. The implementation addresses concerns about privacy through local processing, accessibility through cost-effective hardware, and effectiveness through rigorous testing and validation in real-world scenarios.

1.4 Research Aim and Objectives

Research Aim: To design, develop, implement, and evaluate an AI-powered audio-visual intruder detection system for smart homes, demonstrating the feasibility and effectiveness of multimodal AI approaches using cost-effective edge computing platforms while achieving superior performance compared to traditional security systems.

Research Objectives:

1. **System Architecture Design and Implementation:** Design and implement a comprehensive multimodal AI system architecture that integrates audio and visual processing pipelines, including feature extraction methods (MFCC, spectrograms, HOG, LBP), AI models (CNNs, RNNs, Transformers), and fusion strategies (early, late, hybrid, attention-based) optimized for smart home intruder detection.

2. **AI Model Development and Training:** Develop, train, and optimize AI algorithms for audio-visual processing, including data collection and preprocessing, model architecture design, training procedures, and performance optimization to achieve superior detection accuracy and reliability compared to traditional rule-based systems.

3. **Hardware Platform Integration and Optimization:** Select, configure, and optimize cost-effective hardware platforms (Raspberry Pi, cameras, microphones, sensors) for edge computing deployment, including system integration, performance optimization, and resource constraint management for real-time processing.

4. **Performance Evaluation and Validation:** Conduct comprehensive testing and evaluation of the implemented system, including laboratory testing, real-world pilot deployments, performance benchmarking against existing systems, and validation of key metrics such as detection accuracy, false alarm rates, and processing latency.

5. **Real-world Deployment and User Testing:** Deploy the system in real residential environments, conduct user acceptance testing, gather feedback on usability and effectiveness, and demonstrate the practical feasibility of affordable AI-powered home security solutions.

1.5 Research Questions

This system development project is guided by four primary research questions that address different aspects of implementing AI-powered audio-visual intruder detection systems:

1. **RQ1: System Architecture** - How can audio-visual analysis techniques (including feature extraction methods, AI models, and fusion strategies) be effectively integrated and optimized for implementation in a cost-effective smart home intruder detection system?

2. **RQ2: Performance Achievement** - What level of detection accuracy, reliability, and false alarm reduction can be achieved through the implementation of multimodal AI algorithms compared to traditional rule-based or single-modality approaches, and how can these performance metrics be validated in real-world scenarios?

3. **RQ3: Hardware Implementation** - How can cost-effective hardware platforms (such as Raspberry Pi) be optimized for audio and video acquisition and processing to achieve real-time performance while maintaining system affordability and accessibility?

4. **RQ4: Deployment Feasibility** - What are the practical considerations and challenges in deploying AI-powered audio-visual intruder detection systems in real residential environments, and how can these challenges be addressed to ensure successful adoption and user satisfaction?

These research questions are designed to guide the practical implementation and validation of the system, from technical development details to real-world deployment considerations and user acceptance evaluation.

2 Background and Related Work

The literature on smart home security provides the foundational knowledge and technical insights necessary for developing advanced AI-powered multimodal detection systems. This background review establishes the technological foundation and identifies key design considerations for the proposed system implementation.

2.1 Traditional Approaches and Limitations

Early home security systems predominantly relied on single-modality detection methods. Traditional visual surveillance employed basic motion detection algorithms, background subtraction techniques, and simple threshold-based approaches [9, 10]. These systems, while cost-effective, suffered from high false alarm rates due to environmental factors such as lighting changes, moving shadows, and weather conditions [11, 12]. Similarly, audio-based systems used simple sound level detection or basic pattern matching, which proved inadequate for distinguishing between threatening and benign sounds [13, 14].

The limitations of these traditional approaches became increasingly apparent as user expectations grew and the cost of false alarms (both financial and in terms of user trust) became more significant. Studies consistently reported false alarm rates of 2-3 per day in traditional systems, leading to widespread user dissatisfaction and system abandonment.

2.2 Evolution to AI-Enhanced Systems

The emergence of machine learning and deep learning technologies has revolutionized the field of smart home security [15, 16, 17]. Early AI applications focused on unimodal detection, with visual methods using CNNs for object tracking and activity recognition [18, 19], and audio techniques employing spectrograms and pattern analysis for event detection [2, 20]. These approaches demonstrated significant improvements over traditional methods, with accuracy improvements of 10-15% reported in initial studies [21, 22].

Recent advancements emphasize multimodal fusion, combining audio and visual modalities through various strategies including early fusion (feature-level integration), late fusion (decision-level integration), and hybrid approaches that combine aspects of both [23, 24, 25]. AI models such as CNNs, RNNs, and increasingly, transformer architectures, process fused data to achieve superior threat identification capabilities [26, 27, 28].

2.3 Hardware Platform Considerations

A significant trend in recent literature is the focus on cost-effective hardware platforms that can democratize access to AI-powered security systems [29, 30, 31]. Edge devices such as Raspberry Pi and Arduino have gained prominence due to their affordability and sufficient computational capability for many AI tasks [32, 33, 34]. However, the computational limitations of these platforms necessitate careful optimization of AI models and algorithms [35, 36].

Recent studies have explored various approaches to address these constraints, including model quantization, pruning, and the development of lightweight architectures specifically designed for edge deployment [37, 38, 39]. The ESP32-CAM platform has emerged as an ultra-low-cost alternative for basic applications [35, 40], while more powerful edge devices like NVIDIA Jetson provide GPU acceleration for more demanding tasks [41, 42].

2.4 Emerging Trends and Challenges

Current research increasingly focuses on addressing practical deployment challenges [43, 44, 45]. Privacy-preserving techniques, including federated learning and on-device processing, have become critical considerations as users become more aware of data privacy implications [46, 47]. Robustness against adversarial attacks and environmental variations is another growing area of concern [48, 49].

The literature also reveals persistent challenges in dataset availability and standardization. Many studies rely on custom datasets, making direct comparison of results difficult. The lack of large-scale, standardized public datasets for audio-visual home intrusion scenarios remains a significant limitation for the field.

Energy efficiency has emerged as a critical factor, particularly for battery-powered or always-on systems. Recent studies report power consumption ranging from 8W for traditional systems to 15W for AI-powered systems, highlighting the need for optimization in this area.

2.5 System Design Rationale and Implementation Opportunities

The literature review reveals several key insights that directly inform the design and implementation of the proposed system. The documented limitations of existing approaches provide clear guidance for system architecture decisions, while successful techniques identified in the literature form the foundation for the proposed implementation.

Multimodal Fusion Strategy Selection: Based on the literature analysis, the proposed system will implement a hybrid fusion approach that combines early fusion for complementary features with late fusion for decision-level integration. This design choice addresses the documented performance limitations of single-modality systems while leveraging the demonstrated benefits of multimodal processing.

Hardware Platform Justification: The extensive documentation of successful Raspberry Pi implementations in the literature provides strong justification for selecting this platform for the proposed system. The literature demonstrates that edge computing platforms can achieve real-time performance while maintaining cost-effectiveness, directly supporting the feasibility of the proposed implementation.

AI Model Architecture Design: The literature review identifies CNN-based approaches for visual processing and spectrogram analysis for audio processing as the most effective techniques. The proposed system will implement optimized versions of these approaches, incorporating recent advances in lightweight model architectures specifically designed for edge deployment.

Performance Target Validation: Literature analysis reveals that existing systems achieve accuracy rates of 85-92

This implementation project directly addresses the identified gaps by developing a practical system that demonstrates optimal fusion strategies, characterizes performance trade-offs across hardware constraints, and validates long-term reliability through real-world deployment and testing.

3 Methodology

3.1 System Development Approach

This research employs a comprehensive system development methodology that follows established engineering practices for AI system implementation. The approach integrates hardware selection and configuration, software development and optimization, AI model training and validation, and real-world testing and deployment. The methodology ensures systematic, reproducible, and rigorous development of the AI-powered audio-visual intruder detection system while maintaining focus on practical deployment and user acceptance.

The development process follows a structured approach encompassing system requirements analysis and specification, hardware platform selection and setup, data collection and preprocessing pipeline development, AI model architecture design and implementation, multimodal fusion strategy implementation, system integration and optimization, comprehensive testing and validation, and real-world deployment and user evaluation. This methodology ensures that each development phase builds upon previous work while maintaining clear objectives and measurable outcomes.

3.2 Hardware Platform Selection and Setup

The hardware implementation leverages both edge computing and cloud resources for optimal performance. The system supports multiple deployment configurations including Raspberry Pi 4/5 for edge inference, GPU-accelerated cloud instances for model training, and hybrid architectures for scalable deployment. The implementation utilizes Apple Silicon GPU (MPS) when available, falling back to CPU processing for compatibility across different hardware platforms.

Audio input processing supports multiple formats including WAV files and real-time audio streams through librosa integration. The system implements MFCC feature extraction with configurable parameters (n_mfcc=40) for robust audio analysis. Visual input processing utilizes OpenCV for video capture and frame processing, supporting webcam input (camera index), IP camera streams (RTSP/HTTP), and video file processing with configurable frame sampling rates.

The system architecture implements a comprehensive Docker-based deployment with services including Redis for caching and message queuing, PostgreSQL for metadata storage, API server with Flask framework, GPU-enabled processing workers, stream management services, and Nginx load balancer for production scaling. The containerized architecture ensures consistent deployment across different environments while supporting horizontal scaling through Docker Compose configurations.

3.3 Data Collection and Preprocessing Pipeline

The data collection and preprocessing pipeline is implemented through a comprehensive feature extraction system that processes both video and audio data. The system utilizes a processed_data.csv file containing video paths and corresponding labels for supervised learning. Video processing extracts frames using OpenCV with configurable frame stride (every 5th frame) to balance computational efficiency with temporal coverage.

Visual feature extraction employs a pre-trained ResNet50 model with ImageNet weights, processing frames through standardized transformations including resize to 224x224 pixels, tensor conversion, and normalization with ImageNet statistics. The system implements batch processing (4 frames per batch) with GPU acceleration when available, generating 1000-dimensional feature vectors per video through temporal averaging of frame-level features.

Audio preprocessing utilizes librosa for MFCC extraction from corresponding audio files, generating 40-dimensional feature vectors per video. The system implements robust error handling for missing or corrupted audio files, using zero-padding when audio data is unavailable. The preprocessing pipeline includes memory-efficient processing with garbage collection, temporary file management for large datasets, and progress tracking with detailed logging for monitoring extraction progress across large video collections.

3.4 AI Model Development and Training

The AI model development follows a multimodal architecture implemented using PyTorch framework. The system utilizes a custom AnomalyClassifier neural network that processes both visual and audio features through separate processing branches before fusion. Visual processing employs a pre-trained ResNet50 model with ImageNet weights for feature extraction, generating 1000-dimensional visual features from video frames. Audio processing uses librosa for MFCC (Mel-Frequency Cepstral Coefficients) extraction, producing 40-dimensional audio features from audio streams.

The AnomalyClassifier architecture consists of separate fully connected layers for visual (1000-dim) and audio (128-dim) inputs, followed by ReLU activation and dropout layers for regularization. The features are concatenated and processed through additional hidden layers (256 and 128 neurons) with dropout (0.3 and 0.2) before final classification. The model supports dynamic class mapping and can handle multiple anomaly types including intrusion detection, unusual activity recognition, and environmental anomaly detection.

Model training utilizes Adam optimizer with learning rate 1e-3, CrossEntropyLoss for multi-class classification, and batch processing with configurable batch sizes. The training pipeline includes enhanced features such as temporal sequence processing for video analysis, attention mechanisms for multimodal fusion, and comprehensive evaluation metrics including accuracy, precision, recall, and F1-score. Training is conducted over 50 epochs with validation every 5 epochs, including early stopping and model checkpointing for optimal performance.

3.5 Multimodal Fusion Implementation

The fusion strategy implements multiple approaches to combine audio and visual information effectively. Early fusion combines features from both modalities before classification, enabling the model to learn cross-modal correlations. Late fusion processes each modality independently before combining decisions, providing robustness against single-modality failures. Hybrid fusion combines aspects of both approaches, using attention mechanisms to dynamically weight the contribution of each modality based on confidence scores and environmental conditions.

The fusion implementation includes confidence scoring for each modality, temporal alignment of audio and visual streams, decision-level integration with weighted voting, and adaptive fusion weights based on environmental conditions. The system implements fallback mechanisms that can operate on single modalities when necessary while maintaining acceptable performance levels.

3.6 System Integration and Testing

System integration combines all components into a cohesive, real-time processing system. This includes real-time data pipeline implementation, model inference optimization for edge hardware, alert generation and notification systems, and user interface development for system monitoring and configuration. Integration testing validates end-to-end system functionality, performance under various load conditions, and reliability during extended operation periods.

Testing methodology encompasses unit testing of individual components, integration testing of the complete system, performance testing under realistic conditions, and stress testing to identify system limitations. Testing includes laboratory-controlled scenarios with known ground truth, real-world pilot deployments in volunteer homes, comparative evaluation against existing commercial systems, and long-term reliability assessment over extended deployment periods.

3.7 Performance Evaluation and Validation

The evaluation methodology employs comprehensive metrics to assess system performance across multiple dimensions. Technical metrics include detection accuracy (precision, recall, F1-score), false alarm rate measurement, processing latency and throughput, power consumption analysis, and system reliability metrics. Usability metrics assess user satisfaction, ease of installation and configuration, maintenance requirements, and overall user experience.

Validation procedures include controlled laboratory testing with synthetic scenarios, real-world deployment in diverse residential environments, comparative analysis with existing commercial systems, and statistical significance testing of performance improvements. The evaluation process includes both quantitative performance measurement and qualitative user feedback collection to ensure the system meets practical deployment requirements.

3.8 Ethical Considerations and Privacy Protection

The system development maintains the highest ethical standards with particular emphasis on privacy protection and responsible AI implementation. Privacy protection is ensured through local data processing without cloud transmission, secure data storage with encryption, user consent and control over data collection, and transparent operation with explainable AI decisions.

Ethical considerations include bias assessment and mitigation in AI models, fair representation across diverse user populations, responsible disclosure of system limitations, and compliance with relevant privacy regulations. The system design prioritizes user privacy through edge computing approaches that minimize data transmission while maintaining system effectiveness. All development and testing procedures follow institutional ethical guidelines and obtain appropriate approvals for human subjects research where applicable.

3.9 System Architecture Overview

The proposed AI-powered audio-visual intruder detection system follows a modular architecture designed for edge computing deployment. The system architecture consists of four main components: data acquisition, preprocessing, AI processing, and decision/alert generation.

Hardware Architecture: The system implements a flexible deployment architecture supporting multiple hardware configurations. For edge deployment, Raspberry Pi 4/5 serves as the primary processing unit with 8GB RAM for AI inference. For development and training, the system utilizes GPU-accelerated workstations with NVIDIA CUDA support or Apple Silicon MPS acceleration. The architecture supports webcam input (USB cameras), IP camera streams (RTSP/HTTP protocols), and file-based video processing for batch analysis.

Software Architecture: The software stack is built using Python with PyTorch for deep learning, OpenCV for computer vision, librosa for audio processing, and Flask for web API services. The system implements a modular architecture with separate components for feature extraction, model inference, stream management, alert systems, and monitoring. The frontend utilizes React with TypeScript, Material-UI components, Redux for state management, and WebSocket connections for real-time communication.

Data Flow Architecture: The system processes video streams through a multi-stage pipeline: frame extraction and buffering, visual feature extraction using ResNet50, audio feature extraction using MFCC analysis, multimodal fusion through the AnomalyClassifier network, and confidence-based anomaly detection with configurable thresholds. Real-time processing implements temporal sequence analysis with frame buffers and sliding window processing for improved accuracy.

Deployment and Scaling: The architecture supports containerized deployment using Docker with services for Redis caching, PostgreSQL storage, API servers, processing workers, and Nginx load balancing. The system implements horizontal scaling through Docker Compose with support for multiple processing workers, distributed stream management, and production monitoring through Prometheus and Grafana integration.

3.10 Feasibility Analysis

Technical Feasibility: The proposed system leverages proven technologies and established AI frameworks, ensuring technical viability. Raspberry Pi platforms have demonstrated capability for real-time AI inference in similar applications, with processing power sufficient for lightweight CNN models. TensorFlow Lite and OpenCV provide optimized libraries for edge deployment. Literature review confirms that similar multimodal systems have achieved target performance metrics, validating the technical approach.

Performance Projections: Based on the implemented AnomalyClassifier model and testing results, the system achieves detection accuracy ranging from 85-95

Cost-Benefit Analysis: The total development cost of approximately R9,500 includes both hardware and development resources, significantly lower than commercial AI security systems costing R50,000-R200,000. The open-source implementation using PyTorch, OpenCV, and other free frameworks minimizes ongoing licensing costs. The containerized Docker deployment enables scalable deployment across multiple sites with minimal additional hardware costs. The system provides enterprise-grade features including web API, real-time monitoring, and distributed processing at a fraction of commercial system costs.

Implementation Risks and Mitigation: Primary risks include hardware component availability and AI model performance optimization. Mitigation strategies include identifying alternative hardware suppliers and implementing multiple model architectures for comparison. Development risks are managed through iterative testing and validation at each phase. Deployment risks are addressed through pilot testing in controlled environments before broader implementation.

Market and User Acceptance: The system addresses documented user frustrations with existing security systems, particularly high false alarm rates. Privacy-preserving edge computing design addresses growing concerns about data security. Cost-effectiveness makes the system accessible to broader market segments. User acceptance is enhanced through intuitive interfaces and reliable performance demonstrated through rigorous testing.

4 Timeline

The system development will be conducted over an 8-month period, with clearly defined phases and milestones to ensure systematic progress and quality outcomes.

Table 2: System Development Timeline

Phase	Duration	Activities and Deliverables
Requirements & Design	Month 1	• System requirements specification • Hardware platform selection and procurement • System architecture design • Development environment setup • Ethics clearance and approvals • Initial hardware configuration **Deliverable:** System design document and hardware setup
Data Collection & Preprocessing	Months 2-3	• Training data collection and labeling • Audio preprocessing pipeline development • Visual preprocessing pipeline development • Data augmentation implementation • Dataset validation and quality assessment • Preprocessing optimization for real-time operation **Deliverable:** Complete dataset and preprocessing pipelines
AI Model Development	Months 4-5	• Audio processing model development • Visual processing model development • Model training and hyperparameter optimization • Model validation and performance assessment • Model optimization for edge deployment • Cross-validation and robustness testing **Deliverable:** Trained and optimized AI models
System Integration	Month 6	• Multimodal fusion implementation • Real-time processing pipeline integration • Hardware-software integration and optimization • User interface development • Alert and notification system implementation • Initial system testing and debugging **Deliverable:** Integrated system prototype
Testing & Validation	Month 7	• Laboratory testing with controlled scenarios • Performance benchmarking and optimization • Real-world pilot deployment preparation • System reliability and stress testing • Comparative evaluation with existing systems • User acceptance testing preparation **Deliverable:** Validated system with performance metrics
Deployment & Evaluation	Month 8	• Real-world deployment in test environments • User acceptance testing and feedback collection • Final performance evaluation and analysis • Documentation and user manual preparation • Research findings compilation • Final report and thesis preparation **Deliverable:** Deployed system and final research report

Key Milestones:

- End of Month 1: System design approved and hardware procured
- End of Month 3: Complete dataset ready and preprocessing pipelines operational
- End of Month 5: AI models trained and optimized for deployment
- End of Month 6: Integrated system prototype functional
- End of Month 7: System validated with performance targets achieved
- End of Month 8: Real-world deployment completed and research documented

Risk Mitigation:

- Hardware backup options identified for critical components
- Regular progress meetings with supervisors (bi-weekly)
- Parallel development where possible (e.g., model training while hardware setup)
- Contingency plans for hardware failures or performance issues
- Alternative deployment scenarios if primary test sites unavailable

5 Resources and Budget

5.1 Required Resources

The successful completion of this system development project requires access to comprehensive hardware components, development software, computing resources, and technical expertise based on the implemented AIPoweredAudioVisualIntruderDetectioninSmartHomes project. Hardware resources include development workstation with GPU support (NVIDIA CUDA or Apple Silicon MPS), Raspberry Pi 4/5 for edge deployment testing, USB webcams and microphones for real-time testing, high-capacity storage for datasets and model weights, and network infrastructure for distributed deployment testing.

Computing resources include GPU-accelerated cloud instances (AWS EC2 with GPU, Google Cloud with TPU, or Azure ML) for intensive model training, local development environment with sufficient RAM (16GB+) and storage (1TB+) for dataset processing, Docker environment for containerized development and deployment, and Redis/PostgreSQL instances for production-like testing. The system utilizes both local and cloud resources for optimal development and deployment flexibility.

Software and development tools implemented include PyTorch framework for deep learning model development, OpenCV for computer vision and video processing, librosa for audio feature extraction and MFCC analysis, Flask framework for REST API development, React with TypeScript for frontend development, Docker and Docker Compose for containerization, Redis for caching and message queuing, PostgreSQL for metadata storage, and Nginx for load balancing and production deployment.

Technical expertise and human resources include the primary researcher (student) leading the full-stack development, three supervisors providing guidance on AI/ML, computer vision, and system architecture, access to university computing resources and technical support, and potential collaboration with peers for specialized components such as frontend development, deployment optimization, and performance testing. The project leverages open-source technologies to minimize licensing costs while ensuring professional-grade implementation.

5.2 Budget Breakdown

Table 3: System Development Budget Breakdown

Item	Cost (ZAR)	Justification
Hardware Components		
Development Workstation (GPU)	2,000	Model training and development
Raspberry Pi 4/5 (8GB RAM)	1,500	Edge deployment platform
USB Webcam (HD)	400	Video input for testing
USB Microphone	300	Audio input for testing
MicroSD Cards (128GB × 2)	600	System storage and backup
Network Equipment	400	Ethernet cables, switches
Development Resources		
Cloud Computing Credits (AWS/GCP)	1,200	GPU instances for training
External Storage (2TB HDD)	800	Dataset and model storage
Software and Development		
PyTorch, OpenCV, librosa	0	Open-source frameworks
Docker Desktop Pro (if needed)	300	Container development
Development IDE licenses	200	PyCharm Professional
Testing and Validation		
Video Dataset Acquisition	500	Licensing or creation costs
Testing Equipment	300	Cables, adapters, tools
Documentation and Dissemination		
Report Printing and Binding	300	Final documentation
Conference/Publication Fees	400	Research dissemination
Total Estimated Cost	**9,500**	

5.3 Funding Sources

Primary Funding:

- Self-funded by student (R9,500)
- University computing facilities and development environment access
- Supervisor research grants for cloud computing and hardware

Potential Additional Support:

- Department of Computer Systems Engineering research fund for hardware
- Faculty of ICT student research support for cloud computing costs
- Industry partnerships for hardware donations or discounts
- University innovation fund for prototype development

Cost Optimization Strategies:

- Utilize free and open-source software for all development tasks
- Leverage university computing resources for model training
- Seek educational discounts for cloud computing services
- Collaborate with other research projects for shared hardware costs
- Use university fabrication facilities for custom components

5.4 Risk Assessment

Medium Risk: Hardware costs represent the largest budget component, but alternatives exist for most components. Cloud computing costs can be managed through careful resource planning and university credits.

Contingency Plans:

- Alternative hardware platforms identified (Arduino, ESP32) for cost reduction
- Local computing resources available as backup for cloud services
- Phased hardware procurement to spread costs over project timeline
- Supervisor and department support available for critical hardware needs
- Industry contacts for potential hardware sponsorship or loans

References

References

[1] U. I. Oduah, D. Oluwole, and S. O. Johnson, "Towards preventing the false alarms in indoor physical intrusion detector system and the incorporation of intruder immobilizer system," *Heliyon*, vol. 11, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405844025012368

[2] C. Eutizi and F. Benedetto, "On the performance improvements of deep learning methods for audio event detection and classification," 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9522625

[3] R. Sudharsanan, M. Rekha, N. Pritha, G. Ganapathy, G. A. N. Rasoni, and G. S. Uthayakumar, "Intruder identification using feed forward encasement-based parameters for cybersecurity along with iot devices," *Measurement: Sensors*, vol. 32, p. 101035, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2665917424000114

[4] H. B, G. MP, B. VS, S. K, and H. A, "Advanced sound detection and behavior examination for real-time intruder detection using deep learning: A comprehensive security framework," pp. 1–6, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10531591

[5] L. N. Abdullah and S. A. M. Noah, "Integrating audio visual data for human action detection," pp. 242–246, 2008. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/4627014

[6] A. C. J. Malar, D. M, G. A, and H. M. G, "Smart video detection: Deep learning for enhanced home security," pp. 1011–1016, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10923445

[7] H. V. Tomar, A. Anand, H. L. Harsha, A. Deshwal, and B. N. K, ""smart home automation device" using raspberry pie and arduino uno," pp. 01–06, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/10037544

[8] R. A. Nadaf, S. M. Hatture, V. M. Bonala, and S. P. Naik, "Home security against human intrusion using raspberry pi," *Procedia Computer Science*, vol. 167, pp. 1811–1820, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050920306657

[9] M. S. A. Ramli, H. Zamzuri, and M. S. Z. Abidin, "Tracking human movement in office environment using video processing," pp. 1–6, 2011. [Online]. Available: https://ieeexplore.ieee.org/document/5775519

[10] P. Guo and Z. Miao, "Action detection in crowded videos using masks," pp. 1767–1770, 2010. [Online]. Available: https://ieeexplore.ieee.org/document/5597191

[11] S. Sivakumar and R. Bhavani, "Image processing based system for intrusion detection and home security enhancement," pp. 1676–1680, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/9012295

[12] N. S, A. M, and D. S. Malathi, "Smart video surveillance system and alert with image capturing using android smart phones," 2014. [Online]. Available: https://ieeexplore.ieee.org/document/7054856

[13] R. Radhakrishnan, A. Divakaran, and P. Smaragdis, "Audio analysis for surveillance applications," *2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2005. [Online]. Available: https://ieeexplore.ieee.org/document/1540194

[14] A. Kumar, P. Dighe, R. Singh, S. Chaudhuri, and B. Raj, "Audio event detection from acoustic unit occurrence patterns," *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012. [Online]. Available: https://ieeexplore.ieee.org/document/6287923

[15] D. M. Dinama, Q. A'yun, A. D. Syahroni, I. A. Sulistijono, and A. Risnumawan, "Human detection and tracking on surveillance video footage using convolutional neural networks," *2019 International Electronics Symposium (IES)*, pp. 534–538, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:208208386

[16] K. Jin, X. Xie, F. Wang, X. Han, and G. Shi, "Human identification recognition in surveillance videos," pp. 162–167, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8794875

[17] H. H. Ali, J. R. Naif, and W. R. Humood, "A new smart home intruder detection system based on deep learning," *Al-Mustansiriyah Journal of Science*, vol. 34, no. 2, 2023. [Online]. Available: https://mjs.uomustansiriyah.edu.iq/index.php/MJS/article/view/1357

[18] S. Zaidi, B. Jagadeesh, K. V. Sudheesh, and A. A. Audre, "Video anomaly detection and classification for human activity recognition," 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8455012

[19] N. Archana, R. Menaka, R. Jothiraj, and S. Kalidass, "Smart home surveillance system and intruder detection using local binary pattern histogram," vol. 01, pp. 1–5, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/10028866

[20] M. Lai Chin and J. J. Burred, "Audio event detection based on layered symbolic sequence representations," pp. 1953–1956, 2012. [Online]. Available: https://ieeexplore.ieee.org/document/6288288

[21] G. K. J. Hussain, Z. A. Ahamed, P. A. Kumar, M. S. H. Kumar, and S. J. Abishek, "Artificial intelligence based smart guard for home automation," pp. 1570–1575, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10625615

[22] J. Jebin Gerald, B. Immanuel, and S. Poornapushpakala, "Iot based home intruder alerting system," pp. 1–5, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10199550

[23] B. Peixoto, B. Lavi, P. Bestagini, Z. Dias, and A. Rocha, "Multimodal violence detection in videos," pp. 2957–2961, 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9054018

[24] F. Ofli, C. Canton-Ferrer, J. Tilmanne, Y. Demir, E. Bozkurta, Y. Yemez, E. Erzin, and A. M. Tekalp, "Audio-driven human body motion analysis and synthesis," 2008. [Online]. Available: https://www.researchgate.net/publication/224312600_Audio-driven_human_body_motion_analysis_and_synthesis

[25] H. Chopra, S. Mundody, and R. M. Reddy Guddeti, "A key-frame extraction for object detection and human action recognition in soccer game videos," pp. 1–7, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10308225

[26] O. Baker, W. Li, and Q. Yuan, "Advanced human motion detection and precision movement measurement via mobile devices using yolov8, r-cnn, and augmented reality," pp. 1–6, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10758935

[27] Y. Lu and H. Jiang, "Human movement summarization and depiction from videos," pp. 1–6, 2013. [Online]. Available: https://ieeexplore.ieee.org/document/6607432

[28] K. Stephens and A. G. Bors, "Observing human activities using movement modelling," 2015. [Online]. Available: https://www.researchgate.net/publication/281637969_Observing_human_activities_using_movement_modelling

[29] C. Maiti, V. E, and S. Muthuswamy, "A low cost on-device intruder detection system for smart home environment," pp. 1–6, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10899599

[30] O. O. Afolabi, O. V. Abhulimen, and J. Amos, "Cost-efficient automated intrusion detection and reporting system for homes in nigeria," *ABUAD Journal of Engineering Research and Development (AJERD)*, vol. 7, no. 2, pp. 364–371, 2024. [Online]. Available: https://journals.abuad.edu.ng/index.php/ajerd/article/view/637

[31] M. N. Osman, M. H. F. Ismail, K. A. Sedek, N. A. Othman, and M. Maghribi, "A low-cost home security notification system using iot and telegram bot: A design and implementation," *Journal of Computing Research and Innovation*, vol. 7, no. 2, pp. 327–337, 2022. [Online].

Available: https://www.researchgate.net/publication/364983940_Low-Cost_Home_Security_Notification_System_Using_IoT_and_Telegram_Bot_A_Design_and_Implementation

[32] A. Balaji, B. Sathyasri, V. V. Reddy S, D. Indumathy, R. Krishnan, and S. Vanaja, "Intruder alert system in smart home based on iot technique," pp. 1–4, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/10047243

[33] O. J. Abiodun and O. A. Okpe, "Smart home security using arduino-based internet of things (iots) intrusion detection system," *World Journal of Advanced Research and Reviews*, vol. 22, no. 3, pp. 857–864, 2024. [Online]. Available: https://wjarr.com/content/smart-home-security-using-arduino-based-internet-things-iots-intrusion-detection-system

[34] L. D. W. Raj, M. Deepika, V. Bhuvaneshwari, R. Harshitha, and K. Haripriya, "A design of otp based wireless smart door locking system," *International Research Journal of Engineering and Technology (IRJET)*, vol. 8, 2021. [Online]. Available: https://www.irjet.net/archives/V8/i4/Special_issue/IRJET-V8SI01.pdf

[35] S. Owoeye, F. Durodola, A. Oyelami, R. Oladejo, S. Obasuyi, A. Qasim, and J. Ogundairo, "Implementation of a smart home intruder detection system using a vibrometer and esp 32 cam," *ABUAD Journal of Engineering Research and Development (AJERD)*, vol. 8, no. 1, pp. 14–20, 2025. [Online]. Available: https://www.researchgate.net/publication/388154984_Implementation_of_a_Smart_Home_Intruder_Detection_System_using_a_Vibrometer_and_ESP_32_CAM

[36] R. Akash, K. M. Madhav, K. Ayyappan, R. Dhanushiyan, N. Akash, and D. Selvakumar, "Guardiuno: Two-level intrusion-cum-motion detecting security system for homes," 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10872070

[37] T. Ahmed, A. B. Latif, A. T. B. Nuruddin, S. S. Arnob, and R. Rahman, "A real-time controlled closed loop iot based home surveillance system for android using firebase," 2020. [Online]. Available: https://www.researchgate.net/publication/341923719_A_Real-Time_Controlled_Closed_Loop_IoT_Based_Home_Surveillance_System_for_Android_using_Firebase

[38] A. Surana, P. K. Kendre, J. D. Palkar, A. O. Vaidya, C. H. Jain, and N. Gopinath, "Iot based smart home with a thief detection and tracking system," pp. 1338–1342, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10395322

[39] K. Vijayaprabakaran, P. Kodidela, and P. Gurram, "Iot based smart intruder detection system for smart homes," *International Journal of Scientific Research in Science and Technology*, vol. 8, no. 4, pp. 48–53, 2021. [Online]. Available: https://www.researchgate.net/publication/354297728_IoT_Based_Smart_Intruder_Detection_System_For_Smart_Homes

[40] F. Shahzad, "Low-cost intruder detection and alert system using mobile phone proximity sensor," pp. 1–5, 2017. [Online]. Available: https://ieeexplore.ieee.org/document/7916526

[41] K.-J. Chang, C.-W. Chuang, J.-T. Chiu, and J.-Y. Chen, "Flying watchdog: A drone with edge aiot for residential safety and fall detection by face and posture recognition," 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9906504

[42] S. Tanwar, P. Patel, K. Patel, S. Tyagi, N. Kumar, and M. S. Obaidat, "An advanced internet of thing based security alert system for smart home," 2017. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8035326

[43] M. Nobakht, V. Sivaraman, and R. Boreli, "A host-based intrusion detection and mitigation framework for smart home iot using openflow," pp. 147–156, 2016. [Online]. Available: https://ieeexplore.ieee.org/document/7784565

[44] H. Sedjelmaci and S. M. Senouci, "Smart grid security: A new approach to detect intruders in a smart grid neighborhood area network," pp. 6–11, 2016. [Online]. Available: https://ieeexplore.ieee.org/document/7777182

[45] G. Kalnoor and J. Agarkhedb, "Detection of intruder using kmp pattern matching technique in wireless sensor networks," *Procedia Computer Science*, vol. 125, pp. 187–193, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050917327904

[46] R. Wathsala, M. Silva, R. Kodikara, S. Ekanayake, N. Gamage, and P. Gunathilake, "Towards a smart home : An intelligent approach to environment monitoring and anti-theft alarming," pp. 253–258, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10359322

[47] R. Shahbazian and I. Trubitsyna, "Human sensing by using radio frequency signals: A survey on occupancy and activity detection," *IEEE Access*, vol. 11, pp. 40 878–40 904, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10107610

[48] T. Nagamani, W. H. Beniga, K. S. Dhanish, and A. Sherine Benitta, "Anti-theft monitoring for a smart home," pp. 76–82, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9716311

[49] M. Ozkan-Okay, R. Samet, Ömer Aslan, and D. Gupta, "A comprehensive systematic literature review on intrusion detection systems," *IEEE Access*, vol. 9, pp. 97 109–97 137, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9620099