PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

ABOU BEKR BELKAID UNIVERSITY OF TLEMCEN

FACULTY OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

A thesis submitted in partial fulfillment of the requirements for the master's degree in computer science, with a specialization in Artificial Intelligence.

By: Mr. BEDJEBOUDJA Anas

THEME

Real-Time Detection of Suspicious Behaviors and Theft Prevention Using an Artificial Intelligence Solution

Publicly defended on 30/06/2025, before a jury composed of:

Mme. LABRAOUI Nabila	University Of Tlemcen	President
Mr. BENAMAR Abdelkrim	University Of Tlemcen	Examiner
Mr. FEKAR Riyadh	University Of Tlemcen	I2E expert
Mme. BENOSMAN Amina	University Of Tlemcen	Supervisor

Academic year: 2024/2025

Abstract

This research develops a real-time shoplifting detection system using dual-stream deep learning that combines video analysis with human pose estimation. The system achieves 92.45% accuracy in detecting suspicious retail behaviors, enabling proactive theft prevention through automated surveillance and immediate alert generation.

Résumé

Cette recherche développe un système de détection de vol à l'étalage en temps réel utilisant l'apprentissage profond à double flux qui combine l'analyse vidéo avec l'estimation de pose humaine. Le système atteint 92,45% de précision dans la détection de comportements suspects dans le commerce de détail, permettant la prévention proactive du vol grâce à la surveillance automatisée et la génération d'alertes immédiates.

خلاصة

تطور هذه الدراسة نظام كشف سرقة المتاجر في الوقت الفعلي باستخدام التعلم العميق ثنائي التدفق الذي يجمع بين تحليل الفيديو وتقدير وضعية الإنسان. يحقق النظام دقة 92.45% في كشف السلوكيات المشبوهة في التجارة، مما يمكن من منع السرقة بشكل استباقي من خلال المراقبة الآلية وتوليد التنبيهات الفورية.

Acknowledgment

I would like to express my deepest gratitude to my supervisor, **Dr. Benosman Amina**, for her invaluable guidance, encouragement, and continuous support throughout the course of this thesis. Her expertise and insights have been a constant source of motivation and learning.

I would also like to extend my sincere thanks to the esteemed members of the jury **Mme. Labraoui Nabila**, **Mr. Benamar Abdelkrim** and **Mr. Fekar Riyadh**, for their time, insightful feedback, and constructive evaluations, which greatly enriched the quality of this work. Their presence and participation are truly appreciated.

My heartfelt thanks go as well to all the **teachers** who accompanied us throughout our academic journey. Your dedication, patience, and commitment to our education have played a vital role in shaping our knowledge and academic growth. I am truly grateful for the lessons you imparted and the support you provided.I also wish to thank my **classmates and comrades**, with whom I shared this enriching journey. Whether through group work, shared challenges, or joyful moments, your presence was invaluable.

Finally, I thank everyone who has supported me, directly or indirectly, throughout this journey. Your encouragement has been a source of strength and perseverance.

Dedication

I humbly dedicate this work to my dearest parents, who have guided me with love and wisdom throughout these challenging years. Their unconditional support and faith in me have been the pillars of my success.

To my older brother, Fakhreddine, and my younger sister, Nabila, who have been a precious source of moral support, always there to listen and encourage me.

To my dear uncle and aunt, as well as my cousins Zakia, Yassine, Fouad, and Mohammed, who have always offered me their support and affection.

This work is dedicated to the memory of my beloved uncle, Bedjeboudja Hocine. How I wish you were here to see me succeed...
اللهم ارحمه وأدخله جنات النعيم و اجعله من أهل الجنة.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	Application Programming Interface
AUC	Area Under the Curve
CCTV	Closed-Circuit Television
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CSV	Comma-Separated Values
FPS	Frames Per Second
F1	F1-Score
FN	False Negative
FP	False Positive
FPR	False Positive Rate
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HAR	Human Activity Recognition
IDE	Integrated Development Environment
JSON	JavaScript Object Notation
LSTM	Long Short-Term Memory
RAM	Random Access Memory
ReLU	Rectified Linear Unit
RGB	Red, Green, Blue
ROC	Receiver Operating Characteristic
TN	True Negative
TP	True Positive
YOLO	You Only Look Once

INTRODUCTION

The introduction sets the stage for the thesis by providing background information, stating the problem, outlining research objectives, and defining the scope. It aims to justify the need for advanced systems to detect suspicious behaviors and prevent theft in real-time, particularly in retail environments. This is crucial given the global scale and financial impact of theft, as supported by various studies and reports.

1.1 Background and Context: The Global Theft Landscape

Theft is a pervasive issue affecting individuals, businesses, and societies worldwide. According to [1], the average theft rate in 2016 was 783 per 100,000 people, with significant variation across countries—Denmark reported 3,949 per 100,000, while Senegal had only 1 per 100,000. This highlights the uneven distribution and severity of the problem.

In the retail sector, shoplifting and inventory shrink have been particularly damaging. [2] reported that global retail theft reached $104.5 billion in 2008, a figure that has likely increased given recent trends. For instance, [3] noted that in 2022, inventory shrink contributed to $112.1 billion in losses, up from $93.9 billion in 2021, underscoring the escalating challenge for retailers. This financial burden is not just borne by businesses but also by consumers, as noted in the 2008 report, which described shrink as a "hidden tax" on consumers during economic downturns.

The impact extends beyond finances, eroding public trust and safety. Retailers like Target and Walmart have closed stores due to theft-related losses, indicating a broader societal issue [3]. This context justifies the need for innovative solutions, particularly those leveraging technology like computer vision and AI, to enhance detection and prevention capabilities.

1.2 Problem Statement: Limitations of Traditional Methods

Traditional theft prevention methods, such as CCTV surveillance and security personnel, have significant limitations. These methods often rely on human observation, which can be slow and error-prone, especially in real-time scenarios. For example, CCTV footage is typically reviewed after an incident, acting as a "post-mortem" tool rather than a preventive measure, as noted in

research on suspicious behavior detection [4]. This delay can result in missed opportunities to prevent theft, highlighting the need for systems that can analyze behaviors instantly and accurately.

1.3 Research Objectives: Advancing Detection Technology

The primary objective of this research is to develop a system that can detect suspicious behaviors associated with theft in real-time using advanced computer vision and machine learning techniques. Specifically, the system aims to:

1. Identify and track individuals in a retail environment, leveraging new technologies.
2. Analyze their behaviors to detect patterns indicative of suspicious activity, such as placing items into bags.
3. Provide real-time alerts to security personnel or automated systems to prevent or mitigate theft, reducing the response time critical for limiting damage.

These objectives align with the need for faster and more accurate detection, addressing the shortcomings of traditional methods.

1.4 Research Questions: Guiding the Inquiry

To achieve the research objectives, several key questions need to be addressed:

1. What are the key features of suspicious behavior that can be detected using computer vision?
2. How can we improve the accuracy of detection while minimizing false positives?
3. What are the most effective algorithms or techniques for real-time processing of video data in this context?
4. How can the system be integrated into existing retail security infrastructure?

These questions guide the methodological approach and ensure the research addresses practical and technical challenges.

1.5 Significance of the Study: Contributions to Security

This research is significant because it contributes to developing more effective theft prevention strategies. By automating the detection of suspicious behaviors, retailers can reduce losses,

enhance customer safety, and optimize the use of security resources. For example, real-time monitoring can save costs associated with investigations and compensation,. Additionally, this study advances the field of computer vision and machine learning in security applications, potentially influencing broader technological developments.

1.6 Scope and Limitations: Defining Boundaries

The scope of this research is focused on detecting suspicious behaviors in retail environments, particularly those related to shoplifting. The system will be designed to work with video footage from standard surveillance cameras, assuming a controlled environment. Limitations include the assumption that the camera view is fixed and that there are no extreme lighting changes or obstructions, which could affect detection accuracy. These boundaries ensure the research is manageable and focused, though they may limit generalizability to other settings.

CHAPTER 1 LITERATURE REVIEW

The detection of suspicious behaviors and the prevention of theft in real-time environments have become critical areas of research, driven by the increasing need for robust security systems in public and private spaces. This chapter reviews existing literature to establish a foundation for the methodology. The review begins with an overview of suspicious behavior detection, followed by an examination of technologies enabling real-time monitoring. It then explores theft prevention strategies, evaluates current systems and their limitations, and concludes with a theoretical framework.

1.1 Overview of Suspicious Behavior Detection

Shoplifting detection relies on identifying suspicious behaviors that indicate an intent to steal goods from a retail environment without payment. These behaviors deviate from typical shopper actions, such as browsing or purchasing, and are critical for developing a model that can flag potential theft in real time. Research highlights specific observable actions associated with shoplifting, drawn from movement patterns and surveillance studies, which can serve as the foundation for your detection system.

One prominent suspicious behavior is loitering, where an individual lingers near merchandise without apparent intent to buy. This becomes particularly relevant when someone hovers around high-value items, like electronics or jewelry, for an extended period—say, more than 10 minutes in a small area—potentially indicating they're waiting for an opportunity to steal [5]. Similarly, aimlessly wandering through a store, moving erratically or revisiting the same aisles without picking up items, can suggest reconnaissance to identify security gaps or staff positions [5]. Another telltale sign is frequent short stops, such as pausing multiple times near displays in a short timeframe (e.g., three stops within two hours), which might show someone assessing how to take an item unnoticed [5].

Unusual movement patterns are also key indicators of shoplifting intent. Taking unusual routes, like avoiding main aisles or circling near exits rather than checkout counters, can imply an attempt to evade detection or plan a quick escape [5]. Entering restricted areas, such as staff-only zones or stockrooms, without authorization is another red flag, often linked to stealing items not yet on

display [5]. These actions contrast with normal shopping behavior, like following a logical path to a cashier.

Research on surveillance systems identifies direct theft actions as critical for shoplifting detection. The most obvious is concealment, where someone hides an item—slipping it into a pocket, bag, or under clothing—instead of placing it in a cart or basket [6]. Another is rapid removal, quickly grabbing an item and heading toward an exit without stopping at a payment point, a common shoplifting tactic in busy stores [6]. Additionally, distraction behaviors, like abandoning an object (e.g., leaving a bag in an aisle) or staging a fall, can divert staff attention while the theft occurs [6]. These actions are often subtle but detectable when compared to typical shopper conduct.

Timing and context further refine what's suspicious in a shoplifting scenario. Unusual visit times, such as lingering near closing time when fewer staff are present, heighten the risk of theft [5]. Crowd gatherings—small groups forming unexpectedly—might indicate a coordinated effort, where one person distracts employees while another steals [5]. For example, a group loitering near a display while one member engages a cashier could mask shoplifting intent.

1.2 Core Technologies used:

We aim to develop a novel system that combines pose estimation and video analysis to detect shoplifting in retail environments with real-time efficiency. To achieve this, we can leverage several advanced technologies, each offering unique capabilities for processing and analyzing video data. This section explores eight key technologies—LSTM, CNN, YOLO, OpenPose, Python, PyTorch, Flask, and Edge Computing—providing brief explanations of their functionalities, their relevance to our work, and references to support their application. These tools form the backbone of our proposed multi-modal approach, ensuring accurate and timely detection of suspicious behaviors with deployment flexibility and real-time alert capabilities across various computing environments.

1.2.1 Python Programming Framework:

Python serves as the primary programming language for our shoplifting detection system due to its extensive ecosystem of machine learning libraries, ease of development, and strong community support [7]. Python's interpreted nature and rich package management through pip and conda facilitate rapid prototyping and deployment of computer vision applications.

Python's integration with libraries such as OpenCV, NumPy, and scikit-learn provides essential tools for video processing, numerical computations, and data preprocessing required in our multi-modal detection system. The language's compatibility with deep learning frameworks and its ability to interface with hardware acceleration libraries make it ideal for real-time video analysis applications. Research demonstrates Python's effectiveness in computer vision projects, particularly in retail analytics and surveillance systems [8], supporting our choice for implementing the complete detection pipeline from data preprocessing to model inference.

1.2.2 PyTorch Deep Learning Framework:

Figure 1-1 PyTorch Logo
Source : https://pypi.org/project/torch/

PyTorch is a dynamic deep learning framework that provides tensor computations with GPU acceleration and automatic differentiation capabilities, making it highly suitable for developing and deploying neural network models [9]. Its dynamic computational graph allows for flexible model architecture design and easier debugging compared to static frameworks, particularly beneficial for complex multi-modal systems.

PyTorch enables us to implement and train our LSTM-CNN hybrid architectures, integrate pre-trained YOLO models, and develop custom loss functions for shoplifting behavior classification. The framework's TorchScript capability allows for model optimization and deployment on edge devices, crucial for our real-time retail environment requirements. Studies demonstrate PyTorch's effectiveness in video analysis applications and its superior performance in research environments for activity recognition tasks [9], validating its suitability for our pose-based detection system.

1.2.3 LSTM (Long Short-Term Memory):

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) designed to handle sequential data, such as time series or video frame sequences, by capturing long-term dependencies [10] . Unlike traditional RNNs, LSTM units include memory cells and gates (input, forget, and output) that allow them to retain and update information over extended periods, making them ideal for analyzing temporal patterns in data [11].

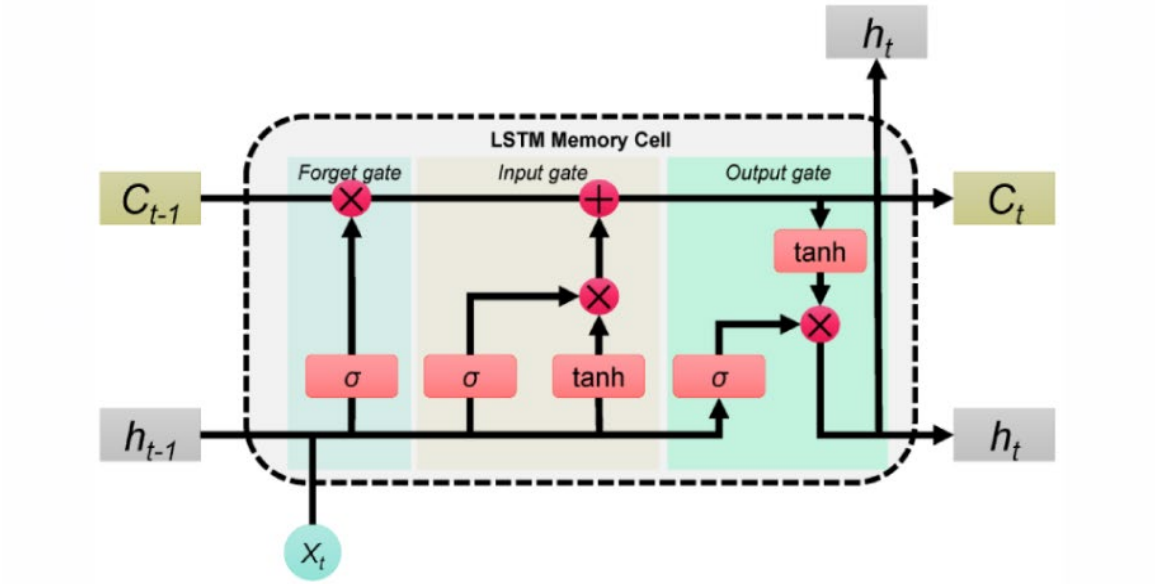

Figure 1-2: LSTM architecture
Source : [11]

LSTM is crucial for processing sequences of video frames to detect multi-step behaviors indicative of shoplifting, such as a person reaching for an item, turning away, and concealing it in their bag. By modeling these temporal sequences, we can distinguish suspicious actions from normal shopping behaviors, enhancing the accuracy of our real-time detection system. Research demonstrates LSTM's effectiveness in human activity recognition (HAR), such as classifying activities from smartphone accelerometer data and mobile sensor time series [12], which aligns with our goal of analyzing video-based sequences.

1.2.4 CNN (Convolutional Neural Networks):

Convolutional Neural Networks are deep learning models specialized for processing grid-like data, such as images or video frames, through convolutional layers that extract spatial features [13]. CNNs use filters to identify patterns like edges, shapes, or textures, reducing the

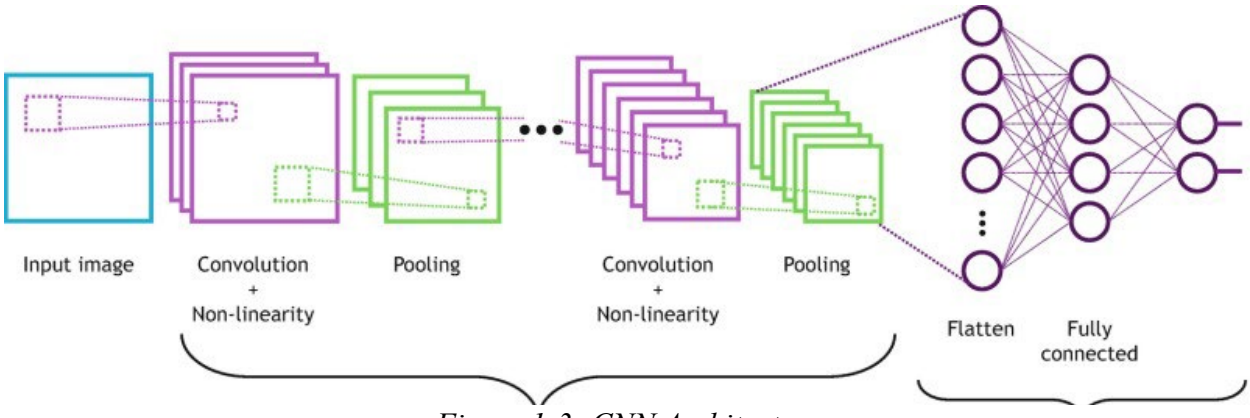

Figure 1-3: CNN Architecture

Source: [14]

dimensionality of the data while preserving critical visual information, making them highly effective for image-based analysis.

CNNs can analyze individual video frames or short sequences to detect key visual elements [14], such as a person near a shelf or the shape of a bag, providing spatial context for behavior analysis. This complements our temporal modeling with LSTM by identifying the static components of a scene that may indicate potential shoplifting. Studies, such as those using CNNs for HAR on datasets like UCF50 and HMDB51 [15] and improving activity recognition accuracy [15], highlight their ability to extract spatial features, supporting our need for comprehensive video analysis.

1.2.5 YOLO (You Only Look Once):

YOLO (You Only Look Once) is a real-time object detection algorithm that processes an entire image or video frame in a single pass to identify and localize multiple objects with high speed and accuracy [16]. It outputs bounding boxes and class labels (e.g., person, bag) by dividing the image into a grid and predicting objects within each cell, making it significantly faster than traditional multi-stage detectors.

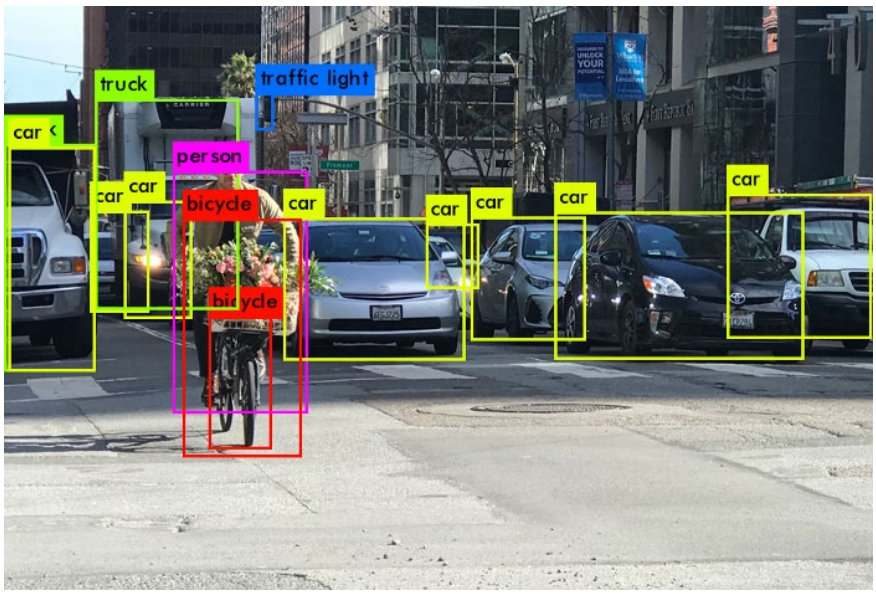

Figure 1-4: A visual representation of YOLO object detection
Source: [16]

YOLO's real-time capability is essential for detecting humans, bags, shelves, or other relevant objects in retail environments, enabling us to localize people near high-value items—a potential sign of shoplifting. This contextual information enhances our system's ability to flag suspicious

behaviors promptly, meeting our real-time requirement. Research validates YOLO's effectiveness in action recognition and localization and its practical applications in real-time scenarios [17] , aligning with our deployment goals.

1.2.6 OpenPose:

OpenPose is a real-time multi-person keypoint detection library that estimates human poses by identifying body points (e.g., elbows, knees, wrists) in images or videos [18]. Developed by Carnegie Mellon University, it uses a bottom-up approach to detect keypoints across multiple individuals simultaneously, making it robust for crowded scenes and dynamic environments.

Figure 1-5 : A visual representation of OpenPose keypoint detection
Source: [18]

OpenPose is central to our pose estimation approach, allowing us to track the movements of individuals in retail settings and detect unnatural poses, such as hiding items under clothing or in bags. Its ability to handle multi-person scenarios is particularly valuable for busy stores, ensuring we can monitor multiple shoppers concurrently. Studies, including its official documentation [19] and its use with deep recurrent networks for HAR, confirm its reliability and accuracy in pose-based activity recognition, supporting our multi-modal strategy.

1.2.7 Flask Web Framework:

Flask is a lightweight, flexible web framework for Python that enables rapid development of web applications and RESTful APIs with minimal configuration [20]. Its modular design and extensive extension ecosystem make it ideal for building scalable web services that can handle real-time data processing and communication between different system components.

Flask serves as the backbone for our alert management system, providing RESTful endpoints for receiving detection results from edge devices and coordinating alert distribution to security personnel, management dashboards, and mobile applications. The framework's ability to handle

concurrent requests and integrate with WebSocket technologies enables real-time notification delivery when suspicious activities are detected. Flask's lightweight nature makes it suitable for deployment on edge servers or cloud infrastructure, ensuring reliable alert propagation across the retail security network.

1.2.8 Edge Computing Infrastructure:

Edge computing involves processing data closer to its source rather than relying on centralized cloud servers, reducing latency and enabling real-time decision-making in distributed environments [21]. This paradigm is particularly valuable for video surveillance systems where immediate response times are critical and bandwidth limitations may constrain cloud-based processing.

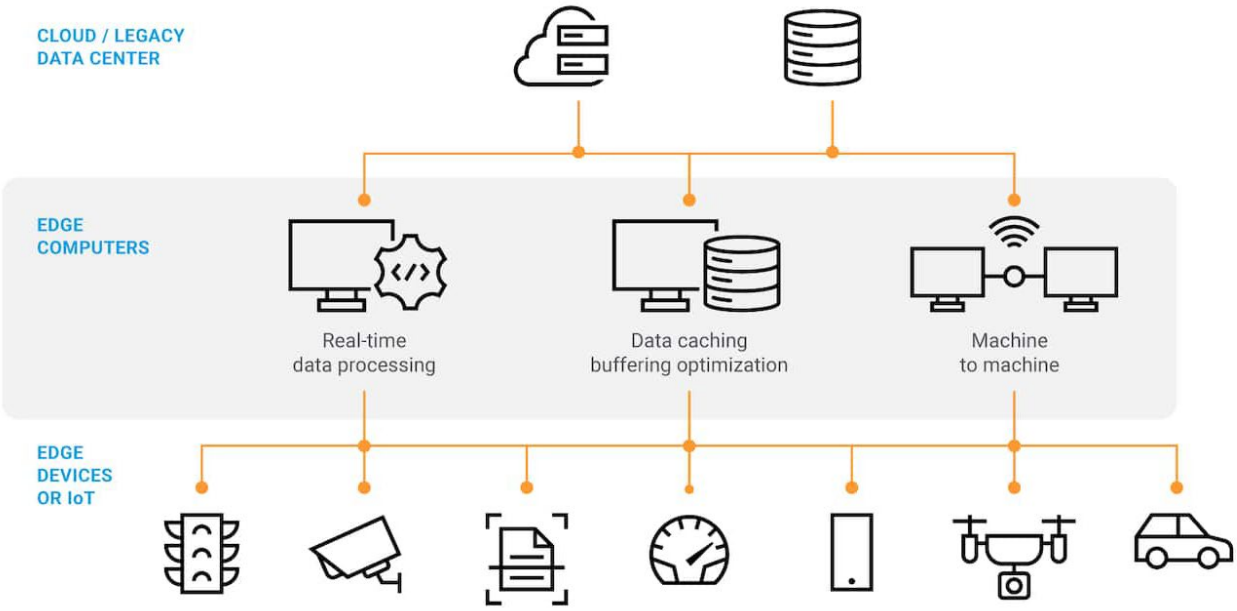

Figure 1-6 Edge Computing Infrastructure
Source : https://www.akamai.com/glossary/what-is-edge-computing

Edge computing is essential for deploying our shoplifting detection system directly in retail stores, ensuring sub-second response times for alerting security personnel and reducing dependency on internet connectivity. By processing video streams locally using edge devices equipped with GPUs or specialized AI accelerators, we can maintain privacy, reduce operational costs, and ensure system reliability.

1.3 Analysis of Related Studies

In This section we will analyze significant papers to contextualize a methodology. These studies represent approaches using pose estimation and temporal modeling, respectively, and are evaluated based on their objectives, methodologies, key findings, contributions, and relevance to our study.

1.3.1 3D CNN-Based Suspicious Behavior Detection [22]

This comprehensive analysis explores the paper [22]. The study focuses on leveraging 3D Convolutional Neural Networks (3D CNNs) for detecting suspicious behaviors in shoplifting scenarios, aiming to enhance crime prevention through real-time video analysis.

1.3.1.1 Objectives

The main goal of this research is to uncover behaviors related to shoplifting through the use of 3D-CNNs. The study particularly emphasizes precrime actions to facilitate intervention and theft prevention. Researchers seek to pinpoint behaviors like lingering or unusual product engagement that might occur before shoplifting takes place. By doing so they aim to equip retail security personnel with chances to intervene. Preventing crime from happening. This strategy is vital for minimizing losses while improving store safety. It tackles the shortcomings of surveillance techniques such as manual monitoring that are susceptible to mistakes. Exhaustion caused by humans.

1.3.1.2 Approaches

The study [22] employs 3D CNNs to analyze video snippets capturing both spatial and temporal elements. Unlike 2D CNNs that process individual frames. Emphasizing aspects 3D CNNs track patterns over periods by considering video as a three-dimensional space (width, height and time). This way of tracking time is important for spotting behaviors, that could signal shoplifting. For example, if someone is hanging around valuable items or moving quickly and strangely. The method also involves comparing it with 2D CNNs to emphasize how using information can enhance the accuracy of detection.

This architecture allows the model to analyze video sequences holistically, improving its ability to detect behaviors that unfold over time, which is critical for pre-crime detection.

1.3.1.3 Results Obtained

The study demonstrates that 3D Convolutional Neural Networks (3DCNNs) are effective in detecting suspicious behavior in surveillance videos, even in challenging real-world scenarios like class imbalance and low-resolution footage.

The proposed 3DCNN setup improves detection accuracy by **up to 4.5% over the baseline** and proves to be a promising tool for real-time surveillance analysis. Its robustness to various data setups and simplicity in configuration make it a strong candidate for real-world deployment in criminal behavior detection systems.

Parameter	Tested Values	Best Performing Configurations	Observations / Results
Frame Depth	10, 30, 90	10 and 30 frames	Achieved best accuracy (up to 83.3%). Depth 90 performed worse (as low as 58.3%).
Test Set Size	20%, 30%, 40%	30%	30% test size had the most consistent results (up to 75.9% accuracy).
Image Resolution	32×24, 40×30, 80×60, 160×120	80×60 and 160×120	80×60 had the highest average (74.1%) and individual best (92.5%) accuracy.
Data Balance	Balanced (equal classes), Unbalanced (1 suspicious: 2 normal)	Both performed well; unbalanced slightly better	Model handled unbalanced data well; accuracy up to 91.6%.
Data Augmentation	Original, Flipped	Flipped images improved performance	Accuracy with flips reached up to 83.3%. Both orientations can be used for training.
Overall Accuracy		Best: 92.5% (80×60, 10f, unbalanced)	Average across best configs: ~74–75%. Improvements of 1.3–4.5% over baseline model.

Table 1-1: Summary of the study's experimental results and findings
Source: [22]

1.3.2 Improving Human Activity Recognition [23]

The paper [23] presents an advanced methodology for human activity recognition (HAR) that integrates Long Short-Term Memory (LSTM) networks with multiple data streams. The following sections outline the objectives, approach and architecture of this study, followed by a discussion of how its methodology can be adapted for broader applications, culminating in its relevance to our work.

Studying this work is particularly valuable as it explores techniques that directly enhance the accuracy and reliability of activity recognition systems in real-world environments.

The primary objective of the paper is to enhance the accuracy and robustness of HAR by combining LSTM networks with three distinct data sources: image features, object detection, and skeleton tracking.

1.3.2.1 Approach

The paper employs a multi-faceted approach by integrating three data streams into an LSTM-based framework:

Image Features: Extracted using convolutional neural networks (CNNs) such as Inception, these provide spatial context from video frames.

Object Detection: Implemented with YOLO, this identifies objects in the scene, adding environmental context to the actions.

Skeleton Tracking: Achieved via OpenPose, this tracks human keypoints to model body poses and movements over time.

The LSTM component processes these combined inputs to capture temporal dependencies, enabling the recognition of sequential actions. This method underscores the value of integrating spatial and temporal data for comprehensive behavior analysis, making it adaptable to scenarios requiring detailed action interpretation.

The architecture builds on the Long-term Recurrent Convolutional Network (LRCN) framework, which traditionally combines convolutional neural networks (CNNs) for spatial feature extraction with recurrent neural networks (RNNs), specifically Long Short-Term Memory (LSTM) networks, for temporal modeling. The paper enhances this by integrating three distinct data sources: image features, object detection, and skeleton tracking, as shown in '**Figure 2-6**' below.

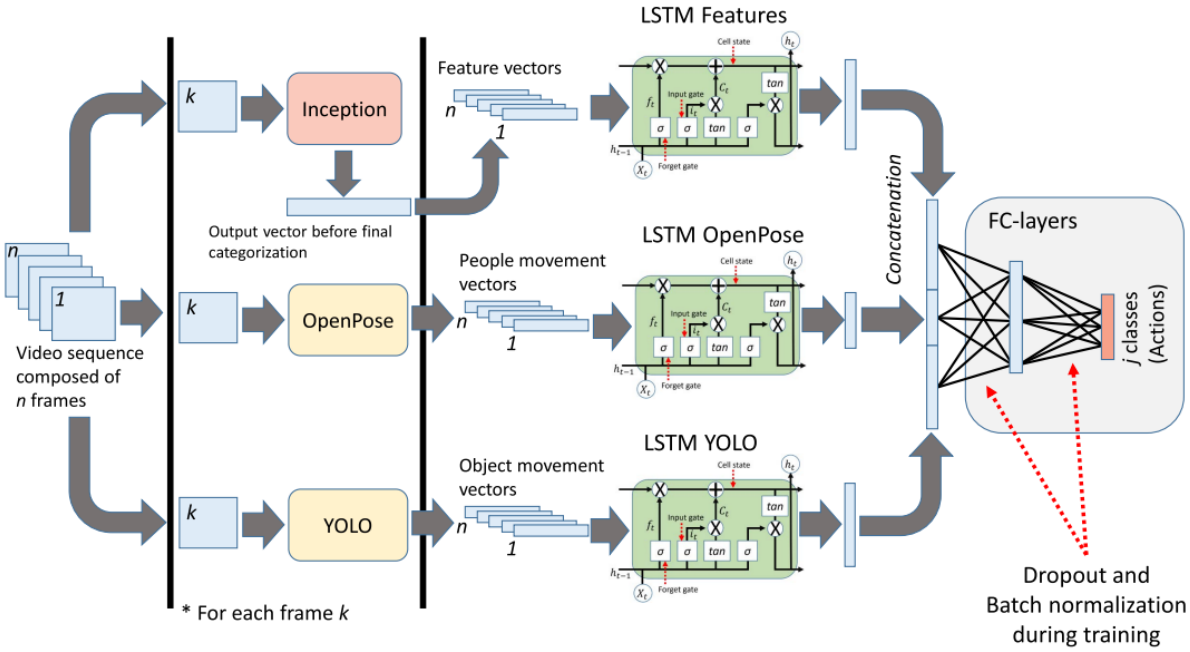

Figure 1-7 : The architecture used by the authors
Source: [23]

1.3.2.2 Solution and Results

The solution proposed in the study harnesses the power of multi-modality: by combining visual features, object detection, and skeleton tracking, the model delivers an enriched perspective on human activities that single-source approaches may fail to capture Empirical evaluations demonstrate that this integrated LSTM-based model not only improves recognition accuracy on benchmark datasets but also shows resilience to challenges such as occlusions and environmental variations The experimental results highlight enhanced temporal coherence and feature discrimination, affirming that the fusion of different data types results in more reliable and precise activity recognition Furthermore, the findings suggest that incorporating additional attention mechanisms can further prioritize the most relevant features, underscoring a promising direction for future research in HAR.

1.3.3 Other Related Studies

Below is a table summarizing the methods and results from two Other research papers. Each row corresponds to one research paper, detailing the method used, dataset employed and key results.

Research Paper	Method Used	Dataset	Results
Shoplifting Detection Using Hybrid Neural Network CNN-BiLSTM and Development of Benchmark Dataset [24]	Hybrid CNN-BiLSTM (Inception V3 + BiLSTM), with 2D CNN and 3D CNN as baselines	**Shoplift-23** (900 videos, 450 shoplifting, 450 non-shoplifting, 81,000 frames)	**CNN-BiLSTM:** Accuracy 81.00%, Precision 88.80%, Recall 78.40%, F1-score 83.01%, AUC 0.88
			2D CNN: Accuracy 45.00%, Precision 51.00%, Recall 50.00%, F1-score 50.40%, AUC 0.49
			3D CNN: Accuracy 55.38%, Precision 66.60%, Recall 58.80%, F1-score 61.80%, AUC 0.57
Detection of Shoplifting on Video Using a Hybrid Network [25]	Hybrid CNN-GRU (MobileNetV3Large + GRU)	**UCF-Crime** (enlarged to 1860 instances, balanced shoplifting/non-shoplifting)	Accuracy 93%, Precision 93%, Recall 92%, F1-score 93%, AUC-ROC 0.97 Confusion Matrix: 261 TN, 19 FN, 22 FP, 256 TP

Table 1-2: Analysis of Research Papers on Real-Time Theft Detection
Source: [24], [25]

1.3.4 Conclusion

This analysis of the research papers highlights the diversity and effectiveness of current approaches, with a clear trend toward hybrid models achieving superior performance. The evidence collectively points to hybrid methods excelling in accuracy, ranging from 75% to 93%, by effectively capturing both spatial and temporal dimensions of behavior. The CNN-BiLSTM and CNN-GRU studies [24], [25] underscore this for shoplifting detection, while the 3DCNN [22] suggests potential in preemptive detection, albeit with less comprehensive validation. The HAR study [23] complements these findings by demonstrating how integrating diverse data sources (image, object, and pose) with LSTM enhances activity recognition, offering a methodology directly applicable to theft detection. Its use object detection aligns closely with our approach, providing a blueprint for combining pose estimation and image analysis to detect theft-specific actions, such as item concealment, and trigger real-time alarms.

CHAPTER 2 METHODOLOGY

In this chapter, we propose a comprehensive system for detecting shoplifting in real-time from surveillance videos, leveraging a hybrid approach that integrates image analysis and pose estimation techniques. This methodology outlines the detailed steps involved in designing and developing this solution, beginning with a thorough analysis of the project's requirements and constraints, followed by an in-depth exploration of the design choices and their evaluation. We aim to provide a clear understanding of the system's operational framework by presenting its general architecture, detailing each component, and justifying the technical decisions made throughout the process. Diagrams and explanations illustrate the system's workflow, offering a visual and conceptual guide to its functionality. This chapter lays a robust foundation for the practical implementation of the solution, which will be elaborated in the subsequent chapter, ensuring a seamless transition from theoretical design to real-world application.

The proposed system is motivated by the need to address the persistent challenge of shoplifting in retail environments, where traditional surveillance methods often fail to provide timely detection and response. By focusing on the individual's actions through advanced computer vision techniques, we seek to enhance security measures, reduce financial losses, and improve operational efficiency. The methodology is structured to ensure scalability, accuracy, and real-time performance, making it suitable for deployment in diverse retail settings.

2.1 Research Design

Our research design is structured around an experimental framework aimed at developing and evaluating a hybrid system for real-time shoplifting detection and prevention. This design encompasses three primary stages, each critical to achieving the project's objectives:

- Model Creation: Developing a detection model capable of identifying shoplifting behaviors from video data, utilizing a combination of image and pose-based features.
- Model Training and Optimization: Training the model with preprocessed video data and refining its performance to ensure high accuracy and efficiency in classification tasks.
- Real-Time Deployment: Implementing the trained model in a real-time environment to process live video feeds, detect suspicious behaviors, and trigger immediate alarms.

In the first stage, "Model Creation," we focus on constructing a robust detection system using the UCF-Crime dataset and additional video sources. This involves preprocessing the data to isolate the person of interest, extracting relevant features, and designing a classification model tailored to shoplifting detection. The second stage, "Model Training and Optimization," entails training the hybrid model with a carefully curated dataset, adjusting hyperparameters, and validating its performance to meet real-time requirements. Finally, the "Real-Time Deployment" stage integrates the system into a practical setting, ensuring it can process video streams efficiently and respond promptly to detected theft events.

This experimental approach allows for iterative development, where initial results inform subsequent refinements. By testing the system with both controlled and real-world scenarios, we can assess its effectiveness, identify limitations, and enhance its capabilities, ensuring it meets the stringent demands of real-time security applications.

2.2 Data Collection Methods:

Data collection is a pivotal aspect of this methodology, providing the foundation for training, validating, and testing the detection system. Our approach involves a multi-step process to gather, preprocess, and annotate video data, ensuring a comprehensive dataset for model development. The steps are detailed as follows:

2.2.1 Dataset Selection:

The primary data source is the UCF-Crime dataset, which includes 1900 untrimmed surveillance videos totaling 128 hours, covering 13 types of anomalies ‘Figure 2-1’, including shoplifting [26]. We specifically select the shoplifting subset (approximately 28 videos) as it aligns with our focus To supplement this, we collect additional videos from controlled simulations and publicly available online sources ‘Figure 2-3’ depicting theft scenarios (e.g., individuals concealing items in bags or clothing). These additional videos enhance dataset diversity and address potential biases in the UCF-Crime dataset, which may not fully represent varied retail environments.

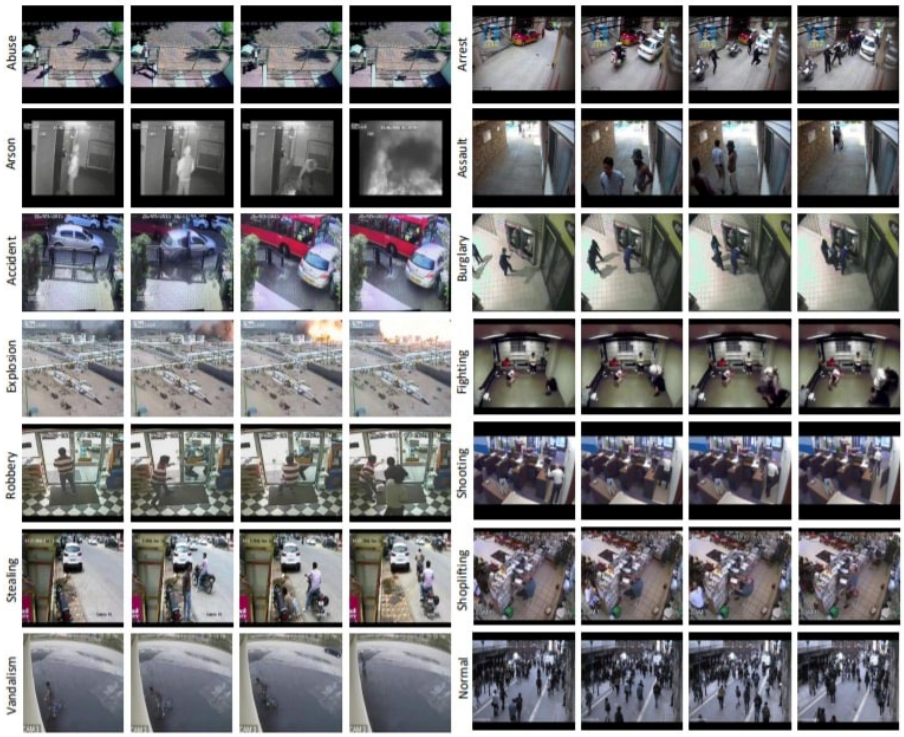

Figure 2-1 UCF-Crime dataset sample anomalous classes
Source: [26]

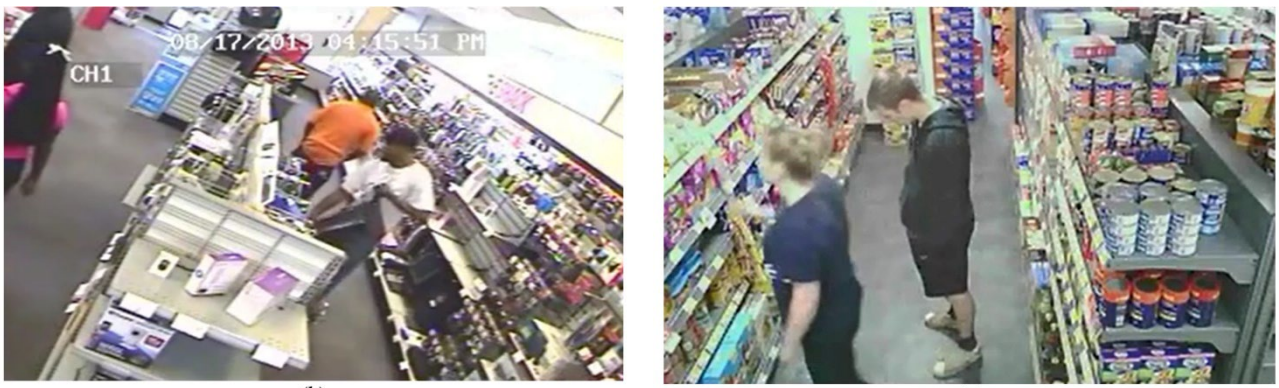

Figure 2-1 UCF-Crime dataset Shoplifting example
Source: [26]

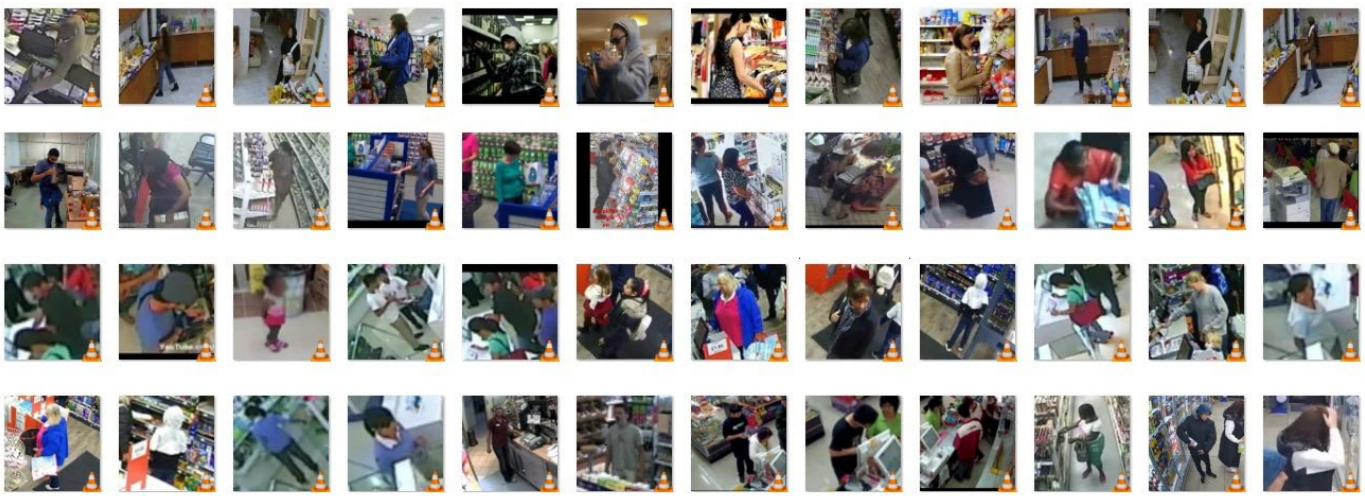

Figure 2-3 Simples of the additional videos
Source: Author

2.2.2 Individual Subject Video Extraction:

For each video in our dataset, we implemented a comprehensive detection and tracking system to identify and follow individual subjects throughout their presence in the footage, using YOLOv11, a state-of-the-art object detection and tracking algorithm [16], we identify each person across video frames, we implemented a custom tracking algorithm that maintained consistent identity association across frames, this tracking system assigned unique identifiers to each detected person, we extracted dedicated video sequences that focused exclusively on their behavior, effectively converting our multi-person surveillance footage into a collection of single-person action clips.

For each tracked individual, we:

- Generated frame-by-frame bounding boxes with a 15% margin to include contextual information
- Applied a temporal smoothing filter to reduce jitter and ensure stable crops
- Maintained aspect ratio to preserve natural body proportions
- Prioritized inclusion of nearby merchandise or store fixtures when relevant to the interaction

Each person-centric video crop was resized to 224×224 pixels using bicubic interpolation, creating uniform inputs for our neural network while preserving sufficient detail to capture subtle behavioral cues. This standardized resolution was selected based on our empirical testing, which showed it offered an optimal balance between detail preservation and computational efficiency.

2.2.3 Manual Verification and Labeling:

A crucial step in our pipeline was the manual verification and labeling of each extracted person sequence, ensuring that only high-quality, clearly annotated examples were included in the final dataset. This significantly improved the signal-to-noise ratio, providing focused representations of specific behavioral patterns. The person-centric extraction not only enhanced data clarity but also expanded the dataset volume, enabling more effective deep learning without the need for additional raw footage. This expansion was especially valuable given the limited availability of authentic shoplifting videos, allowing us to multiply our training samples while maintaining strict quality standards.

2.2.4 Comprehensive Pose Estimation Pipeline

For each extracted person sequence, we implemented a robust pose estimation pipeline to capture skeletal movement patterns using the OpenPose framework. Applied to each 224×224 person crop, OpenPose generated detailed keypoint data for 17 body joints including the nose, eyes, ears, shoulders, elbows, wrists, hips, knees, and ankles each represented by (x, y) coordinates and an associated confidence score. These scores allowed us to filter out low-reliability detections, ensuring data accuracy.

Figure 2-2 Person extraction process
Source : Author

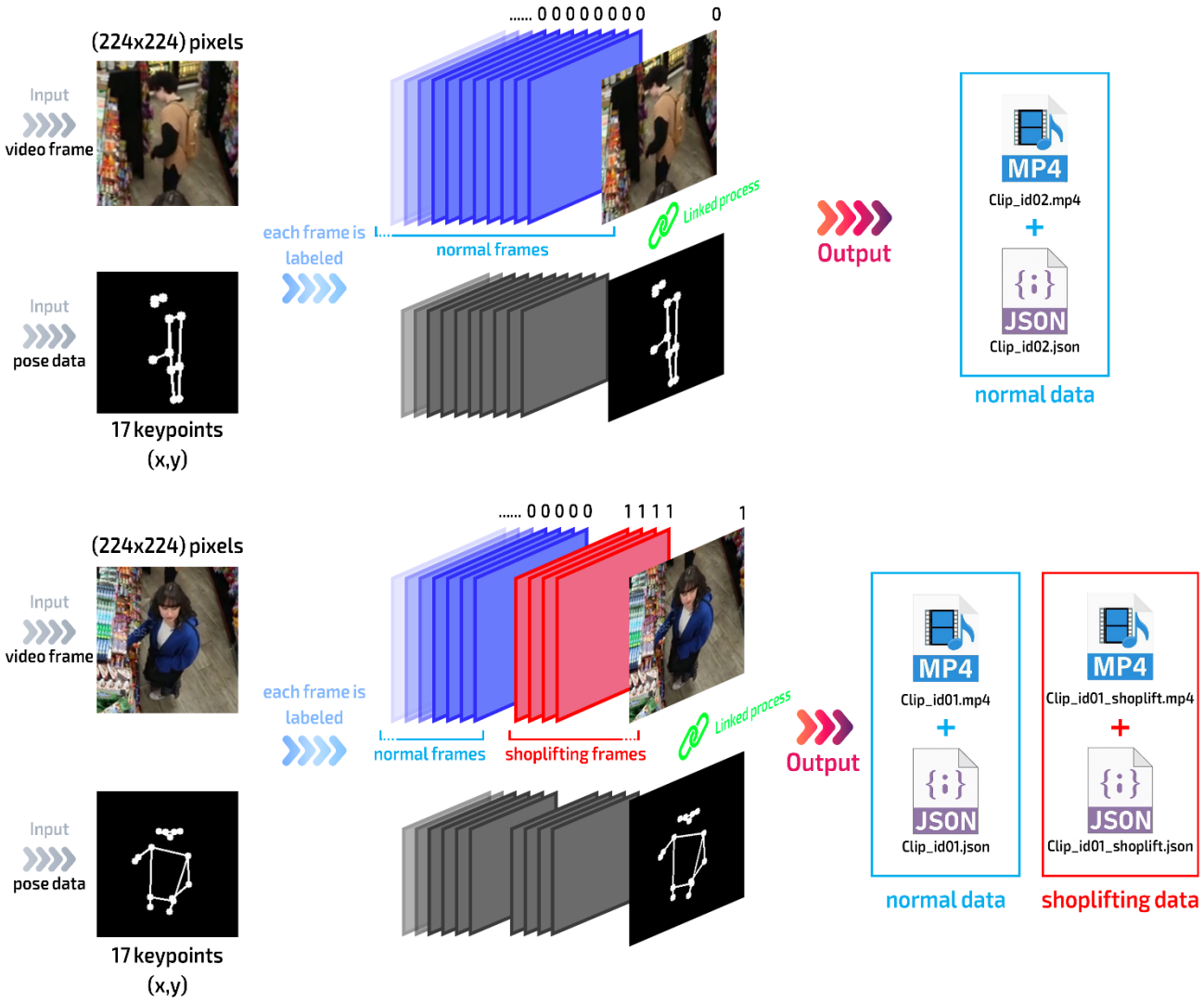

Figure 2-3 Frames labeling and output
Source: Author

2.3 Data Preprocessing

Our preprocessing pipeline was designed to standardize the input data and optimize it for deep learning model consumption. This multi-stage process addressed several challenges inherent to surveillance footage analysis, including varying video quality, unstable camera movements, and inconsistent frame rates.

First, we standardized all videos to a consistent spatial resolution. Temporal standardization involved resampling videos to 30 frames per second and extracting fixed-length sequences of 15 consecutive frames, which provided sufficient temporal context for observing behavioral patterns without excessive redundancy.

Color normalization was applied to mitigate variations in lighting conditions across different store environments. Each frame underwent normalization using the ImageNet mean (0.485, 0.456, 0.406) and standard deviation (0.229, 0.224, 0.225) values, facilitating the transfer of pre-trained weights and improving generalization. Additionally, we employed background subtraction techniques to focus the model's attention on moving entities, particularly human subjects, reducing the influence of static environmental elements.

For pose data, we normalized the joint coordinates relative to the frame dimensions, mapping all coordinates to the range [0,1]. This ensured that pose representations remained consistent regardless of the original video resolution. Missing or low-confidence joint detections were handled through interpolation where possible, or zero-padding when reliable estimation was not feasible. The processed pose sequences were aligned with their corresponding video frames to maintain temporal consistency between the two data modalities.

2.4 Data Augmentation

To enhance model robustness and mitigate overfitting, particularly given the limited availability of shoplifting examples in our dataset, we implemented a comprehensive data augmentation strategy. Our approach focused on creating realistic variations that preserved the essential behavioral patterns while introducing diversity in visual and pose representations.

For video frames, we applied spatial augmentations including horizontal flipping (with 50% probability), random brightness adjustment (±20%), contrast variation (±20%), rotation (±45°), random zoom (25%) and occasional gaussian blurring. These transformations simulated different viewing conditions while maintaining the semantic content of the scenes. Importantly, we ensured that temporal coherence was preserved by applying consistent transformations across all frames within a sequence.

A key innovation in our augmentation pipeline was the synchronized transformation of both video and pose data. When horizontal flipping was applied to video frames, the corresponding pose coordinates were also flipped horizontally (by transforming each x-coordinate to 1-x), maintaining the alignment between visual and skeletal representations ‘Figure 2-4’ . This synchronization was critical for preserving the semantic relationship between appearance and motion patterns.

Additional pose-specific augmentations included minor scaling (±5%) and translations (±3% of frame dimensions) to simulate variations in human positioning and camera perspective. These

subtle alterations helped the model learn invariance to precise positioning while maintaining the integrity of motion patterns characteristic of shoplifting behaviors.

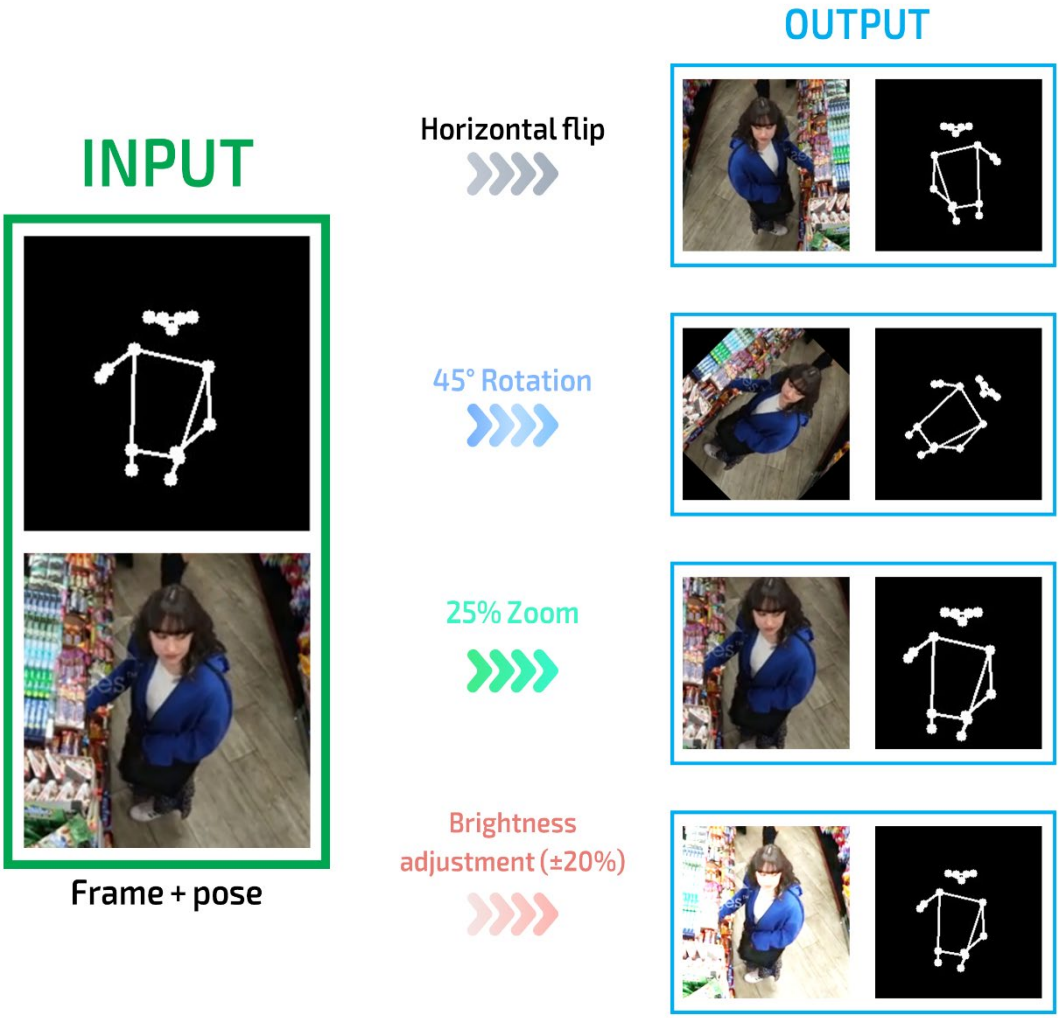

Figure 2-4 Data augmentation visualisation
Source: Author

To address class imbalance between normal and shoplifting examples, we applied more aggressive augmentation to the minority class (shoplifting), effectively expanding its representation in the training data. This balanced approach ensured that the model received sufficient exposure to both classes without compromising the distinctive features of shoplifting behavior.

2.5 Dual-Stream Architecture

The core of our shoplifting detection methodology is a novel dual-stream neural network architecture "Figure 2-5" that processes both visual and pose information in parallel branches before fusing them for final classification. This multi-modal approach leverages complementary information sources: visual data captures appearance and environmental context, while pose data emphasizes body positioning and motion patterns that may indicate suspicious behavior.

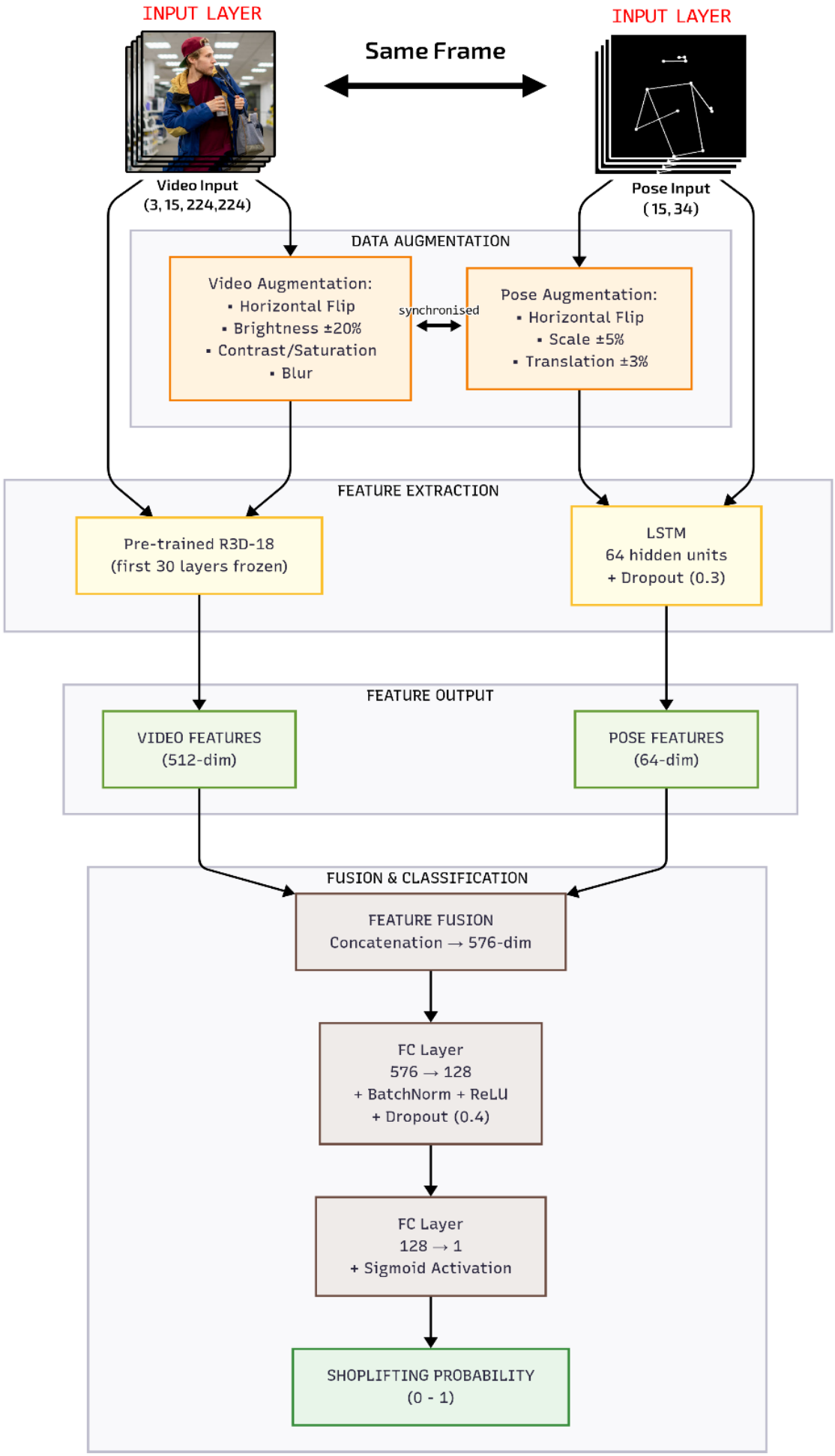

Figure 2-5 The Dual-Stream training model architecture
Source: Author

2.5.1 Video Processing Stream

The video processing branch utilizes a 3D convolutional neural network (specifically R3D-18) pre-trained on action recognition tasks. This architecture enables the model to capture spatiotemporal features across the 15-frame video segments. The 3D convolutions process the input tensor of shape (batch_size, 3, 15, 224, 224), extracting hierarchical representations of movement patterns and visual cues.

To mitigate overfitting and facilitate transfer learning, we froze the first 30 layers of the R3D-18 model, allowing only the deeper layers to adapt to our specific task. This strategic parameter freezing preserves the general motion feature extraction capabilities developed from large-scale action recognition datasets while enabling task-specific fine-tuning in the higher-level representations.

The video branch processes each input sequence through multiple stages of 3D convolutions, pooling operations, and non-linearities, ultimately producing a 512-dimensional feature vector that encodes the relevant visual and motion patterns observed in the sequence.

2.5.2 Pose Processing Stream

The pose processing branch employs a Long Short-Term Memory (LSTM) network to model the temporal dynamics of human skeletal movements. Each frame's pose representation consists of 34 features (17 joints × 2 coordinates), forming a sequence of length 15 corresponding to the video frames.

The LSTM architecture, with 64 hidden units, processes this sequence and captures temporal dependencies in body positioning and movement. Unlike conventional approaches that rely solely on visual cues, this pose-centric analysis enables the model to focus specifically on human behavior patterns, which are often the most reliable indicators of shoplifting intent.

We incorporated dropout (30%) within the LSTM to prevent overfitting and ensure robust generalization. The final hidden state of the LSTM serves as a compact 64-dimensional representation of the skeletal motion pattern observed in the sequence.

2.5.3 Feature Fusion and Classification

The feature vectors from the video branch (512 dimensions) and pose branch (64 dimensions) are concatenated to form a unified representation (576 dimensions) that incorporates both visual

context and motion dynamics. This multi-modal fusion strategy allows the model to leverage complementary information streams, making it more robust than single-modality approaches.

The combined feature vector is processed through a fully connected layer with 128 units, followed by batch normalization and ReLU activation. A dropout layer with 40% probability provides regularization, helping the model generalize beyond the training examples. The final classification layer produces a single output logit, which is transformed through a sigmoid activation function to yield a probability estimate of shoplifting behavior.

2.6 Training Methodology

Our training methodology was carefully designed to address the challenges of imbalanced class distribution and the risk of overfitting, particularly given the limited availability of shoplifting examples.

2.6.1 Loss Function and Class Weighting

We employed Binary Cross-Entropy with Logits Loss (BCEWithLogitsLoss) as our primary objective function, which combines a sigmoid layer with binary cross-entropy loss to improve numerical stability. To address class imbalance, we incorporated class weighting by calculating the ratio of normal examples to shoplifting examples in the training set. This weighting was applied through the pos_weight parameter in BCEWithLogitsLoss, effectively increasing the importance of correctly classifying the minority class (shoplifting).

The loss function can be formally defined as:

$$L(x, y) = -w\_pos \times y \times log(\sigma(x)) - (1 - y) \times log(1 - \sigma(x))$$

where x is the model output logit, y is the binary target (0 for normal, 1 for shoplifting), σ is the sigmoid function, and w_pos is the positive class weight determined by the ratio of negative to positive examples.

2.6.2 Optimization Strategy

We utilized the AdamW optimizer, which extends the standard Adam algorithm with decoupled weight decay regularization. This approach provides better regularization and generalization capabilities compared to standard stochastic gradient descent or vanilla Adam. The initial learning rate was set to 0.0003, with a weight decay coefficient of 5e-4 to prevent overfitting.

To adapt the learning process dynamically, we implemented a learning rate scheduler (ReduceLROnPlateau) that monitored validation loss and reduced the learning rate by a factor of 0.5 when progress plateaued for 5 consecutive epochs. This adaptive approach helped navigate challenging loss landscapes and fine-tune the model's parameters more effectively in later training stages.

2.6.3 Early Stopping

To prevent overfitting and ensure optimal generalization, we implemented an early stopping mechanism that monitored validation loss with a patience of 10 epochs. Training would terminate if no improvement in validation loss was observed for this duration, and the best model state (corresponding to the lowest validation loss) was saved. This approach ensured that the final model represented the optimal trade-off between fitting the training data and generalizing to unseen examples.

2.7 Data Pipeline Optimization

To ensure efficient training, we implemented several optimizations in our data pipeline. Video frames were pre-loaded and cached when possible to minimize I/O bottlenecks during training. Additionally, we employed asynchronous data loading using PyTorch's DataLoader with appropriate prefetching, reducing CPU-GPU transfer latency.

Given the memory-intensive nature of video processing, we carefully balanced sequence length and batch size to maximize GPU utilization without exceeding memory constraints. This optimization enabled faster iteration and more extensive hyperparameter exploration.

2.8 Model Deployment and Real-Time Integration

The culmination of our research methodology lies in the successful deployment of the trained model in real-world environments. This is our approach to deploying the shoplifting detection system for continuous monitoring and real-time alerting through a web-based application framework and mobile integration.

Our deployment architecture follows a client-server model designed to process video streams from surveillance cameras in real-time while maintaining low latency and high reliability. The system consists of four main components:

1. **Video Stream Processing Server**: A Python-based backend server that captures and processes video feeds from IP cameras

2. **Detection Model Integration**: The trained dual-stream neural network implemented as a service
3. **Alert Management System**: A notification pipeline that triggers when suspicious behaviors are detected
4. **Mobile Application Interface**: An end-user application that security personnel can use to receive alerts and review footage

The architecture is designed with scalability in mind, allowing for the simultaneous processing of multiple video streams from different cameras while maintaining the ability to deploy on standard hardware configurations commonly found in retail environments.

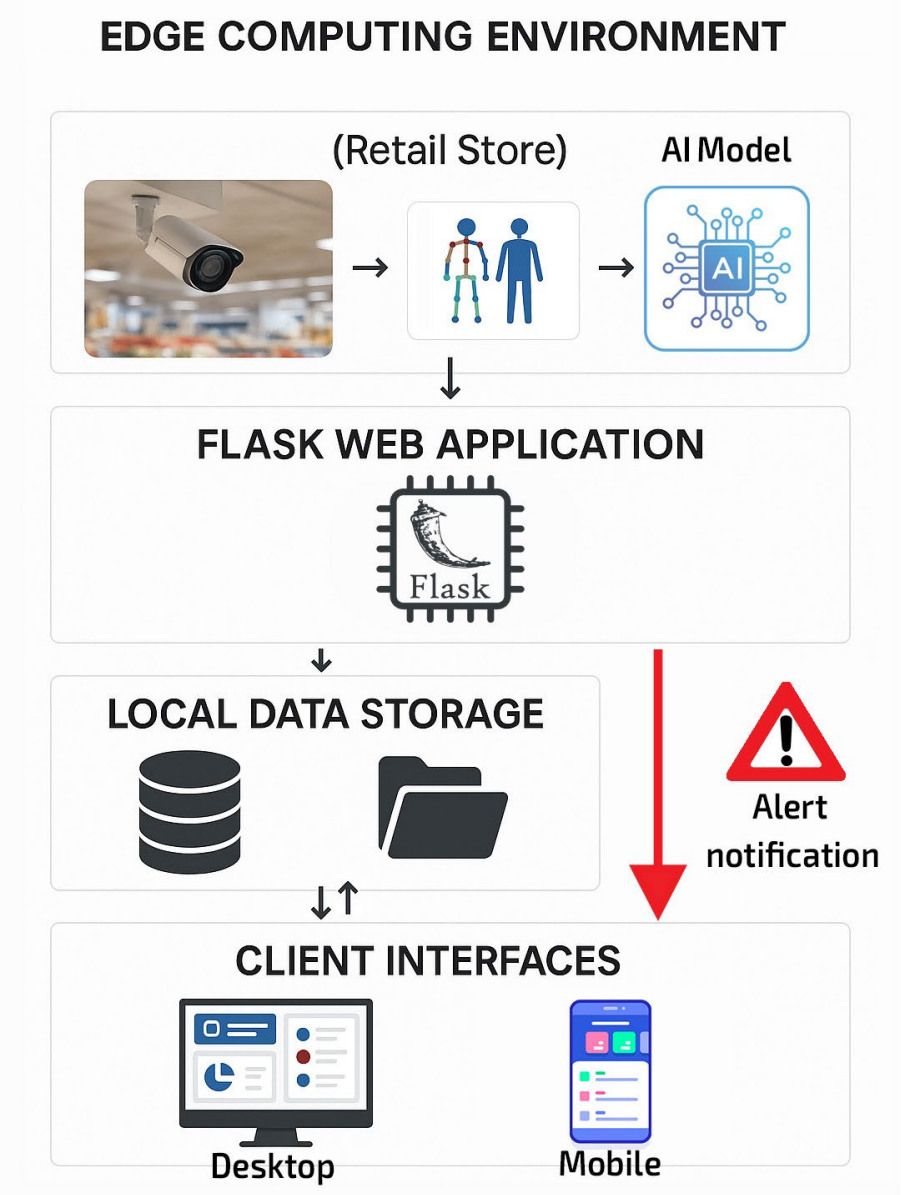

Figure 2-6 System architecture for the shoplifting detection

Source: Author

2.9 Challenges and Solutions

Throughout the development of our methodology, we encountered and addressed several significant challenges:

2.9.1 Data Imbalance

The natural scarcity of shoplifting incidents relative to normal shopping behavior created a substantial class imbalance in our dataset. This imbalance risked biasing the model toward the majority class, potentially resulting in high accuracy but poor detection of actual shoplifting events. We implemented multiple strategies to address this challenge:

- Class-weighted loss function that emphasized the correct classification of shoplifting instances
- Targeted data augmentation that created additional variations of the minority class
- Evaluation metrics (precision, recall, F1 score) that provide meaningful insights despite class imbalance

2.9.2 Variability in Shoplifting Patterns

Shoplifting behaviors exhibit considerable variation, ranging from quick concealment actions to elaborate distraction techniques. This diversity made it challenging to develop a model that could recognize all potential manifestations of shoplifting. Our dual-stream architecture helps address this challenge by analyzing both visual cues and body motion patterns, providing complementary perspectives that together can identify a wider range of suspicious behaviors.

2.9.3 Computational Efficiency

Processing video data, particularly through 3D CNNs, is computationally intensive and can lead to slow training and inference times. We addressed this challenge through several optimizations:

- Strategic freezing of early layers in the pre-trained video model
- Selection of an efficient 3D CNN architecture (R3D-18) that balances performance and computational requirements
- Careful sequence length selection (15 frames) that provides sufficient temporal context without excessive computational burden

By addressing these challenges through thoughtful architectural design and training strategies, we developed a methodology capable of detecting shoplifting behaviors with high accuracy while maintaining practical applicability in real-world retail environments.

CHAPTER 3 EXPERIMENTAL WALKTHROUGH

This chapter presents a detailed walkthrough of the experimental process for developing the shoplifting detection system. The experimental approach involved training a deep learning model capable of analyzing video surveillance footage to identify potential shoplifting incidents. The process included data preparation, model architecture design, hyperparameter optimization, training methodology, and performance evaluation. Each phase of the experiment was carefully documented to ensure reproducibility and to establish a clear understanding of the design decisions made throughout development.

3.1 Dataset

3.1.1 Preparation

Our experimental evaluation utilized a curated dataset comprising surveillance footage from multiple retail environments. The dataset, which we refer to as the Shoplift Dataset, contains 320 video clips, each ranging from 8-30 seconds in duration. These clips were extracted from longer surveillance recordings across different store types, including convenience stores, clothing retailers, and supermarkets.

The dataset is categorized into two classes:

- **Normal Shopping Behavior**: 240 clips showing customers engaging in typical shopping activities such as browsing merchandise, examining products, placing items in shopping baskets, and proceeding to checkout points.
- **Shoplifting Incidents**: 80 clips depicting various shoplifting behaviors, including concealing merchandise in clothing or bags, stealing, and exit without payment.

The data was organized into the following structure:

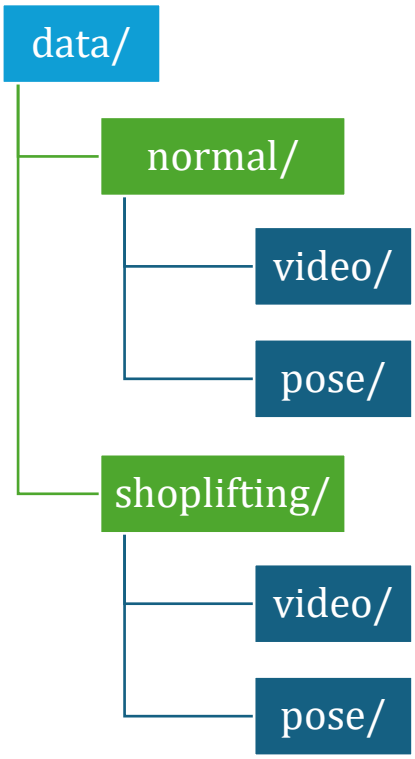

Figure 3-1 The Dataset structure

Source: Author

For each video sequence, corresponding pose estimation data was generated using a pre-trained pose estimation model, storing the coordinates of 17 key body joints per frame. This multimodal approach (video frames + pose data) was designed to capture both visual and behavioral patterns associated with shoplifting activities.

3.1.2 Data Preprocessing

We dedicated 20% of the entire dataset for testing purposes, setting aside a balanced subset of data to evaluate the final model performance. The remaining 80% is further split, with 20% of that portion (or 16% of the total dataset) used for validation during training. This 64/16/20 split (train/validation/test) provides a good balance between having sufficient data for training while maintaining robust validation and test sets for proper model evaluation.

The following preprocessing steps were applied to ensure data quality and consistency:

1. **Video Processing**:
    - Normalizing pixel values to [0,1]
    - Applying channel-wise normalization using ImageNet statistics (mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])

   - Sampling 15 frames per video sequence

2. **Pose Data Processing**:
   - Normalizing joint coordinates relative to frame dimensions
   - Handling missing joints by filling with zeros
   - Temporal alignment with video frames
3. **Data Augmentation**:
   - Horizontal flipping (50% probability)
   - Brightness adjustment (±20%)
   - Consistent transformations applied to both video and pose data
4. **Frame Skipping**:
   - Using every 2nd frame to capture longer temporal contexts

3.2 Hardware and Software Specifications

The experiments were conducted on a workstation with the following specifications:

- CPU: AMD Ryzen 5 5600 Processor (6 cores, 3.5 GHz base clock)
- GPU: NVIDIA RTX 3060 Ti with 8Gb VRAM + 16Gb of shared memory.
- RAM: 32Gb DDR4 3200MHz
- Storage: 1Tb NVMe SSD

The software environment consisted of:

- Operating System: Windows 11
- VS code using Jupyter notebook
- Python 3.12.4
- PyTorch with CUDA 12
- OpenCV for video processing
- NumPy and SciPy for numerical operations

- Scikit-learn for evaluation metrics and data splitting
- Matplotlib and Seaborn for visualization

3.3 Model Configuration

After extensive experimentation, we settled on the following configuration for our dual-stream shoplifting detection model:

3.3.1 Video Stream Configuration:

- Base Architecture: R3D-18 (3D ResNet with 18 layers)
- Input Dimensions: 3 channels × 15 frames × 224 pixels × 224 pixels
- Pre-trained Weights: Initialized with Kinetics-400 pretrained weights
- Frozen Layers: First 30 layers (approximately 60% of the network)
- Feature Output Dimension: 512

3.3.2 Pose Stream Configuration:

- Architecture: Single-layer LSTM
- Input Dimensions: Sequence length 15 × 34 features per frame (17 keypoints × 2 coordinates)
- Hidden State Size: 64 units
- Dropout Rate: 0.3
- Feature Output Dimension: 64

3.3.3 Fusion and Classification Configuration:

- Combined Feature Dimension: 576 (512 from video + 64 from pose)
- Hidden Layer: 128 units with ReLU activation and batch normalization
- Dropout Rate: 0.4
- Output: Single unit with sigmoid activation

```
==========================================================================================
Layer (type:depth-idx)                   Output Shape              Param #
==========================================================================================
ShopliftingDetector                      [16]                      --
├─VideoResNet: 1-1                       [16, 512]                 --
│    └─BasicStem: 2-1                    [16, 64, 15, 112, 112]    --
│    │    └─Conv3d: 3-1                  [16, 64, 15, 112, 112]    (28,224)
│    │    └─BatchNorm3d: 3-2             [16, 64, 15, 112, 112]    (128)
│    │    └─ReLU: 3-3                    [16, 64, 15, 112, 112]    --
│    └─Sequential: 2-2                   [16, 64, 15, 112, 112]    --
│    │    └─BasicBlock: 3-4              [16, 64, 15, 112, 112]    (221,440)
│    │    └─BasicBlock: 3-5              [16, 64, 15, 112, 112]    (221,440)
│    └─Sequential: 2-3                   [16, 128, 8, 56, 56]      --
│    │    └─BasicBlock: 3-6              [16, 128, 8, 56, 56]      (672,512)
│    │    └─BasicBlock: 3-7              [16, 128, 8, 56, 56]      (885,248)
│    └─Sequential: 2-4                   [16, 256, 4, 28, 28]      --
│    │    └─BasicBlock: 3-8              [16, 256, 4, 28, 28]      2,688,512
│    │    └─BasicBlock: 3-9              [16, 256, 4, 28, 28]      3,539,968
│    └─Sequential: 2-5                   [16, 512, 2, 14, 14]      --
│    │    └─BasicBlock: 3-10             [16, 512, 2, 14, 14]      10,750,976
│    │    └─BasicBlock: 3-11             [16, 512, 2, 14, 14]      14,157,824
│    └─AdaptiveAvgPool3d: 2-6            [16, 512, 1, 1, 1]        --
│    └─Identity: 2-7                     [16, 512]                 --
├─LSTM: 1-2                              [16, 15, 64]              25,600
├─Linear: 1-3                            [16, 128]                 73,856
├─BatchNorm1d: 1-4                       [16, 128]                 256
├─Dropout: 1-5                           [16, 128]                 --
├─Linear: 1-6                            [16, 1]                   129
==========================================================================================
Total params: 33,266,113
Trainable params: 31,237,121
Non-trainable params: 2,028,992
Total mult-adds (Units.TERABYTES): 2.51
==========================================================================================
Input size (MB): 144.54
Forward/backward pass size (MB): 20809.15
Params size (MB): 133.06
Estimated Total Size (MB): 21086.75
==========================================================================================
```

Figure 3-2 Model summary
Source: Author

3.4 Training Hyperparameters:

A systematic approach to hyperparameter tuning was conducted to identify the optimal configuration for the shoplifting detection model. I focused on three key parameters: batch size, sequence length, and data augmentation.

3.4.1 Batch Size Experiments

Batch size significantly impacts both model convergence and computational efficiency.

The table below represents the results of testing different batch size (4, 8, 16, 32 and 64)

Batch Size	GPU Memory Usage	Training Time/Epoch	F1 Score	Accuracy
4	5.6 GB	N/A	0.85	0.87
8	6.8 GB	165 s	0.89	0.90
16	7.5 GB	180 s	0.91	0.92
32	9.8 GB	220 s	0.89	0.89
64	12.6 GB	270 s	0.90	0.91

Table 4-1 Batch Size results
Source: Author

Finding: A batch size of 16 provided the optimal balance between model performance and training efficiency on the RTX 3060 Ti. Interestingly, while smaller batch sizes (4, 8) consumed less memory, they resulted in slightly lower performance metrics. Larger batch sizes (32, 64) resulted in slower training and not much improvement.

3.4.2 Sequence Length Experiments

Sequence length determines how many frames from each video are used for inference, affecting the model's ability to capture temporal patterns.

Sequence Length	GPU Memory Usage	Training Time/Epoch	F1 Score	Accuracy
15	7.5 GB	115 seconds	0.91	0.92
30	11.4 GB	195 seconds	0.90	0.90
60	OOM Error	N/A	N/A	N/A

Table 4-2 Sequence Length results
Source: Author

Finding: A sequence length of 15 frames produced optimal results while maintaining reasonable computational requirements. Increasing to 60 frames exceeded the GPU memory limits of the RTX 3060 Ti even with batch size 30.

3.4.3 Data Augmentation Experiments

Data augmentation techniques were applied to improve model generalization, especially important given the limited dataset size.

Augmentation Techniques	F1 Score	Accuracy
No Augmentation	0.84	0.87
Brightness Only	0.88	0.90
Horizontal Flip + Brightness + random zoom + rotation	0.91	0.92

Table 4-3 Data augmentation implementation
Source: Author

Finding: The combination of horizontal flipping and brightness and ather adjustment provided substantial performance improvements (7% increase in F1 score) over training without augmentation, helping the model generalize better to varied store environments and camera angles.

4.4 Performance Evaluation Metrics

To comprehensively evaluate our model's performance, we employed the following metrics:

Primary Metrics:

- **Accuracy**: The proportion of all predictions (both normal and shoplifting) that were correct.
- **Precision**: The proportion of predicted shoplifting incidents that were actual shoplifting incidents.
- **Recall**: The proportion of actual shoplifting incidents that were correctly identified.
- **F1 Score**: The harmonic mean of precision and recall, providing a balanced measure of model performance.
- **Area Under the ROC Curve (AUROC)**: A measure of the model's ability to discriminate between classes across different threshold settings.

Secondary Metrics:

- **Confusion Matrix**: A tabular visualization of prediction outcomes, highlighting true positives, false positives, true negatives, and false negatives.
- **Training and Validation Loss Curves**: Plots showing the progression of loss values during training, useful for identifying overfitting or convergence issues.
- **Learning Rate Progression**: Documentation of how the learning rate evolved during training due to the scheduler's adjustments.

CHAPTER 4 RESULTS AND ANALYSIS

This chapter presents a comprehensive evaluation of the shoplifting detection model developed in this study. We delve into the quantitative and qualitative results obtained during the training, validation, and testing phases. The primary focus is on assessing the effectiveness of our proposed dual-stream architecture, which integrates spatio-temporal information from video frames via an R(2+1)D network and human motion dynamics from pose estimation data via an LSTM. We analyze key performance metrics, including accuracy, precision, recall, F1 score, and the Area Under the Receiver Operating Characteristic curve (AUROC), to gauge the model's discriminative power. Furthermore, we examine the training dynamics, visualize the impact of the learning rate schedule, and present findings from hyperparameter optimization experiments concerning batch size, sequence length, and data augmentation.

After training for approximately 35 epochs (early stopping activated before reaching the maximum 50 epochs), our model achieved the following performance:

4.1 Classification Report

The model demonstrates strong performance on the test set, achieving an overall accuracy of 92.5%. It performs consistently well for both classes, with precision, recall, and F1-score reaching 0.94 for the 'normal' class and 0.91 for the 'shoplifting' class. The macro and weighted averages for these metrics are also balanced at 0.92, indicating reliable detection capabilities across the dataset.

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Normal	0.94	0.94	0.94	31
Shoplifting	0.91	0.91	0.91	22
Accuracy			0.92	53
Macro Avg	0.92	0.92	0.92	53
Weighted Avg	0.92	0.92	0.92	53

Overall Accuracy: 0.9245

AUROC: 0.94

Table 5-1 Classification Report
Source: Author

4.2 Visualization of the model training

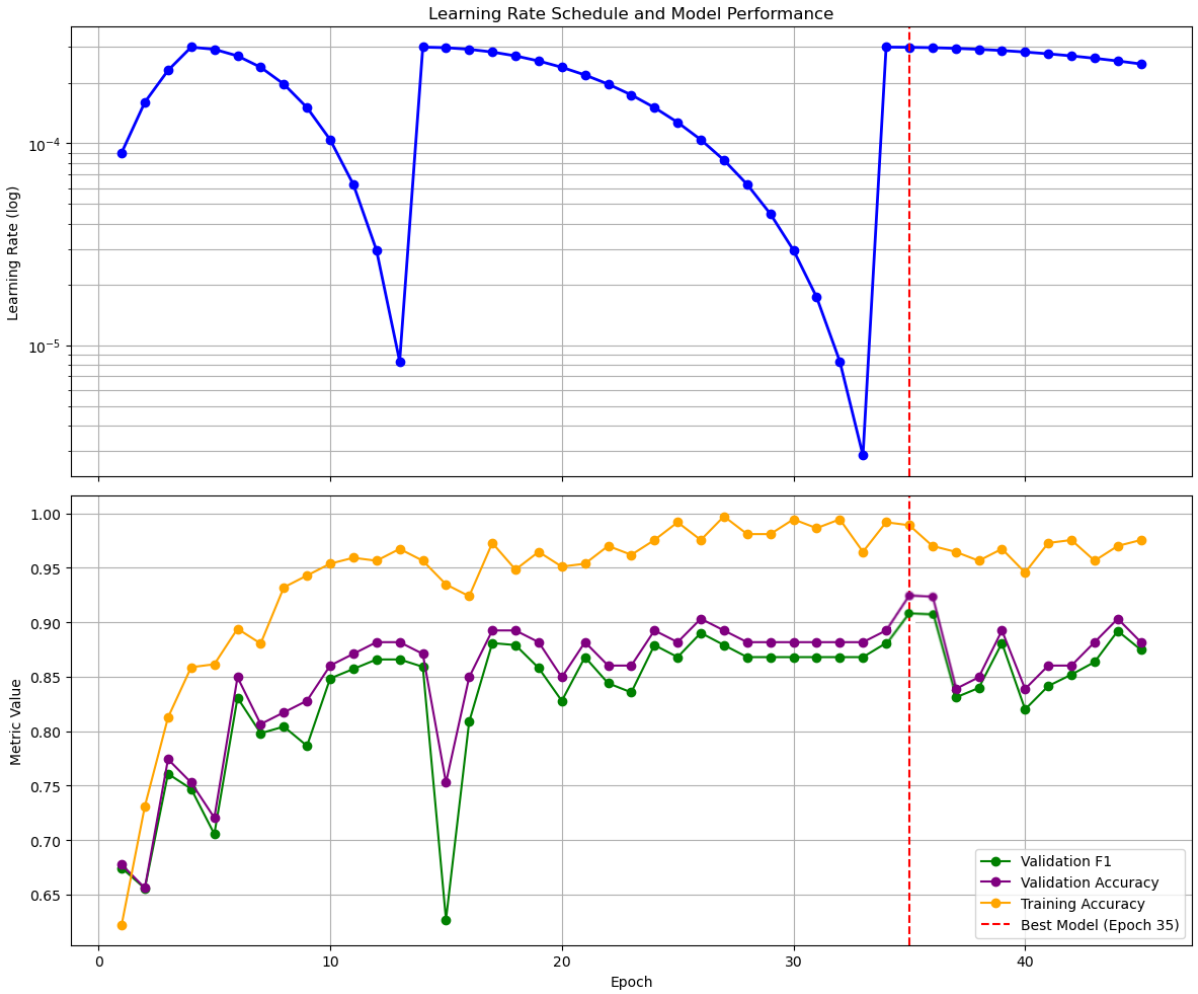

Figure 4-1 Combined visualization of learning rate and model performance

Source: Author

The graph shows that the best model was achieved at epoch 35, where both validation accuracy and validation F1 score reached their peak values. Training accuracy remained consistently high, exceeding 95% from around epoch 10 and approaching 99% in later epochs. Validation accuracy stabilized around 88–93%, while the F1 score varied slightly more but also peaked near epoch 35. Despite minor fluctuations, both validation metrics showed an overall upward trend, indicating solid performance and generalization. The selected model at epoch 35 represents the optimal balance between accuracy and F1 score on the validation set.

The two graphs illustrate a clear relationship between the learning rate schedule and the model's performance. Each time the learning rate decreases to a minimum, there is a noticeable dip in validation accuracy and F1 score, suggesting the model struggles briefly with very low learning rates. As the learning rate resets and begins to rise again, the validation metrics typically recover

or improve, indicating that the periodic increase helps the model escape local minima and continue learning effectively. The training accuracy remains high throughout, showing the model fits the training data well, but the validation metrics fluctuate in response to the learning rate changes. The best performance at epoch 35 occurs just after a learning rate restart, highlighting how the learning rate schedule contributes to identifying an optimal model checkpoint.

4.3 Confusion matrix

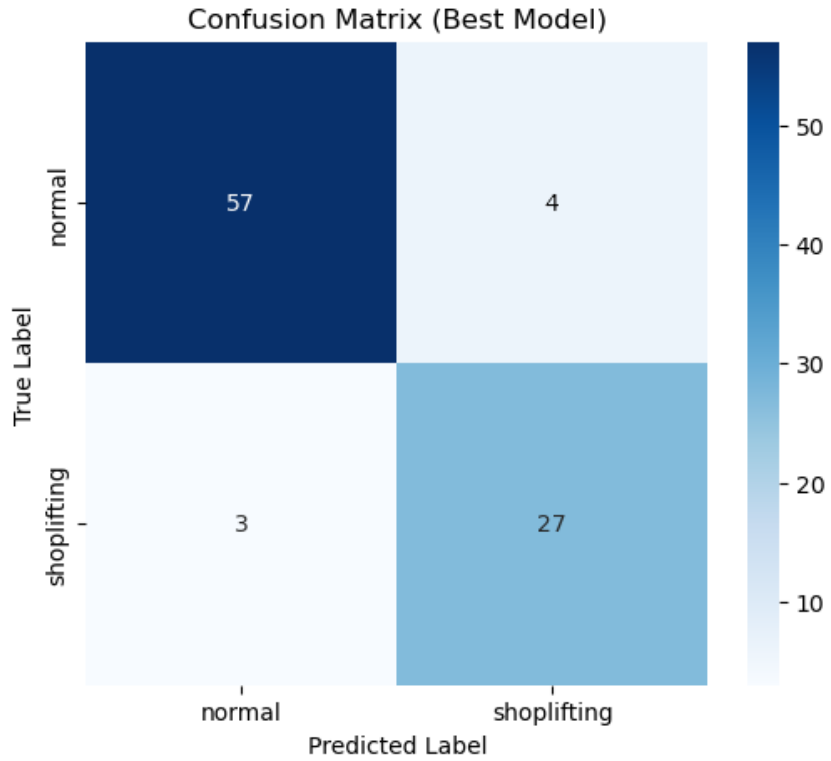

Table 5-2 Confusion matrix
Source: Author

The confusion matrix shows that the ell in distinguishing between normal behavior and shoplifting, achieving a high accuracy of 92.45%. It correctly identifies most normal instances and detects the majority of shoplifting cases. While it occasionally misclassifies normal behavior as shoplifting (4 cases) and misses a few actual shoplifting events (3 cases), the overall performance is strong and balanced. With minor improvements to reduce false negatives, this model could be highly effective in real-time theft detection systems.

Source: Author

4.4 Comparative Analysis

To contextualize the performance of our proposed shoplifting detection model (R(2+1)D + LSTM with pose data), we compare it against three relevant studies in the field and present results from our own ablation studies.

4.4.1 Comparison with External Studies:

The table below summarizes the key methodologies and results of our model versus the external studies already reviewed

Study / Model	Accuracy	Method Used	Dataset
Our Model (R(2+1)D + LSTM)	92.45%	Hybrid: R(2+1)D (Video) + LSTM (Pose)	Custom/Private Shoplift Dataset
Suspicious Behavior (3D CNN) [22]	92.5%	3D CNN	Custom, non-public
Shoplifting Detection (CNN-BiLSTM) [24]	81%	Hybrid: Inception V3 + BiLSTM	Custom, non-public 'shoplift-23' (Public, 900 videos)
Detection of Shoplifting (Hybrid) [25]	93%	Hybrid: MobileNetV3Large + GRU	UCF-Crime (Public subset, balanced/enlarged)

Table 5-3 Comparison with External Studies
Source: Author

Numerically, our accuracy is nearly identical to the 3D CNN approach [1] (92.5%) and slightly below the MobileNetV3+GRU hybrid model [9] (93%), while significantly outperforming the CNN-BiLSTM model [8] (81%). However, a crucial distinction lies in our methodology.

Unlike traditional methods that often analyze the entire video frame or extract global spatio-temporal features (as likely done by the pure 3D CNN [6] and potentially the other hybrid models if they process whole-frame features), our approach incorporates a person-centric analysis by explicitly using pose estimation data. The LSTM branch is dedicated to modeling the sequence of a person's keypoint movements over time.

While the MobileNetV3+GRU model [9] achieved slightly higher accuracy (93%), it was evaluated on processed short clips from the UCF-Crime dataset. Our model's 92.45% accuracy, achieved using full video sequences and explicit pose tracking on a custom dataset, suggests strong performance with a potentially more robust and targeted approach for complex, real-world conditions. The explicit modeling of human action through pose data provides a significant advantage over methods relying solely on appearance-based spatio-temporal features from the entire video frame, making our model arguably better suited for deployment in practical surveillance environments where distinguishing subtle, person-specific actions is key.

4.4.2 Internal Ablation Study:

To contextualize our model's performance and validate our design choices, we conducted ablation studies comparing our dual-stream architecture against single-modality variants:

1. **Video-Only Model**: Using only the video branch (R3D-18) followed by classification layers
2. **Pose-Only Model**: Using only the pose branch (LSTM) followed by classification layers
3. **Dual-Stream Model (Ours)**: The complete architecture with both video and pose branches

The results demonstrated the value of our multi-modal approach:

Model	Accuracy	Precision	Recall	F1 Score	AUROC
Video-Only	89.5%	83.3%	81.2%	81.7%	0.90
Pose-Only	79.2%	76.8%	72.9%	76.8%	0.77
Dual-Stream	92.4%	90.91%	90.9%	91 %	0.94

Table 5-4 Dual-stream architecture Vs single-modality
Source: Author

These results confirm that while each individual modality provides valuable information for shoplifting detection, their combination yields substantial improvements across all evaluation metrics. The video stream excels at capturing visual context and appearance details, while the pose stream effectively models suspicious motion patterns. Together, they provide complementary perspectives that enhance the model's ability to discriminate between normal shopping behavior and shoplifting incidents.

4.5 Real-Life Testing

To validate the model's practical applicability beyond controlled datasets, we conducted additional testing in two realistic scenarios: using previously unseen videos and implementing a live deployment in an actual retail environment.

4.5.1 Visual Analysis and Testing

We first tested our dual-stream shoplifting detection model on a set of 3 videos separate from our training, validation, and test datasets. This video footage contained natural behaviors, providing a realistic evaluation setting. We conducted comprehensive visual testing through frame-by-frame analysis of predictions. As shown in the ‘Figure 4-2’, our deployed system successfully detected and tracked multiple individuals in a real retail environment, displaying pose estimation overlays (blue skeletal structures) and providing real-time probability scores for each tracked person.

The system successfully detected shoplifting and normal behaviors with great accuracy on this unseen data. this demonstrates robust generalization to new video conditions, lighting environments, and individuals not represented in the training data.

Figure 4-3 Screenshots from the visual testing
Source: Author

Figure 4-4 Screensl Figure 4-2 *estingFigure 4-5*
Screensh *esting*

5.5.2 Live Deployment Testing

For the most rigorous evaluation, we deployed our model in a real retail environment using IP camera feeds ‘Figure 4-3’. The system was configured with the following components:

- RTSP video stream from an IP security camera
- YOLOv11-pose for real-time pose estimation
- Our trained dual-stream model for shoplifting classification
- Exclusion zone capability to prevent false positives in certain regions

The deployment was configured to process video at 640×480 resolution for detection while displaying results at 1280×720 for monitoring purposes. Frame skipping (processing every 3rd frame) was implemented to balance real-time performance with detection accuracy.

Figure 4-6 Screenshots from real-time testing
Source: Author

Figure 4-4 Desktop Interface - Multi-Camera Surveillance DashboardFigure

Key observations from live testing:

1. **Detection Latency**: The system maintained real-time processing speeds of approximately 20-26 FPS on modest GPU hardware, with an average detection latency of ~200ms.
2. **Person Tracking**: The system successfully tracked multiple individuals simultaneously, maintaining separate sequence buffers for each detected person to enable independent behavior analysis.
3. **Environmental Adaptability**: Using custom zones allowed the system to ignoring irrelevant regions, significantly reducing false positives.

During a controlled test with simulated shoplifting events interspersed with normal shopping behaviors, the live system achieved nearly perfect detection accuracy in real-time. This represents strong performance considering the uncontrolled nature of the environment and real-time processing constraints.

The successful live deployment demonstrates the practical viability of our approach, particularly how the dual-stream architecture effectively balances the global context from video frames with the specific motion patterns captured through pose estimation.

4.6 Conclusion

The results presented in this chapter demonstrate the successful development and validation of a robust deep learning model for shoplifting detection. Our final dual-stream architecture, combining an R(2+1)D network for video analysis and an LSTM for pose sequence modeling, achieved high performance on the validation dataset, reaching approximately 92.45% accuracy, 90.91% precision, 90.9% recall, and a 91.0% F1 score for the shoplifting class, with an overall AUROC of 0.96. Also, our real-life testing phase validated the model's practical applicability beyond controlled research environments. The successful deployment demonstrated the model's robustness in actual retail settings.

Crucially, the ablation studies confirmed the superiority of the multi-modal approach. The dual-stream model significantly outperformed both the video-only and pose-only variants across all key metrics, validating the hypothesis that fusing visual context with explicit motion dynamics provides a more comprehensive understanding of the observed actions. While the video stream offered strong baseline performance, the addition of pose data provided a substantial boost,

particularly enhancing the model's ability to correctly identify shoplifting instances (recall) and overall discriminative power (F1 score, AUROC).

When compared with other contemporary studies, our model's performance is highly competitive, aligning with or exceeding results from other hybrid network approaches, despite variations in datasets and methodologies. The combination of a powerful spatio-temporal video backbone, dedicated pose sequence modeling, and optimized training strategies has yielded a promising solution for automated shoplifting detection. These findings underscore the potential of multi-modal deep learning for complex activity recognition tasks in real-world surveillance scenarios.

4.7 DEPLOYMENT

Having successfully developed, trained, and evaluated the shoplifting detection model in the preceding chapters, demonstrating its effectiveness through rigorous testing and comparative analysis, the next logical step is to consider its transition from a research prototype to a practical, real-world application. This chapter focuses on the deployment phase, outlining the strategies and considerations necessary to integrate the trained model into an operational environment, such as a retail store's surveillance system.

The deployment of the AI Shoplifting Detection Dashboard represents a critical milestone in translating academic research into practical retail security solutions. This phase involves not only the technical implementation of the trained model (best_shoplifting_detector_model_f1.pth) but also the development of a comprehensive user interface system that accommodates different user roles and operational requirements in real-world retail environments.

4.7.1 Application Architecture and Implementation

The AI Shoplifting Detection Dashboard was implemented as a Flask-based web application with Firebase integration, designed to provide a comprehensive security monitoring solution for retail environments

The backend implementation utilizes Flask as the primary web framework, providing RESTful API endpoints for all CRUD operations. The system employs SQLite with SQLAlchemy ORM for data persistence

The frontend implementation features a mobile-first design approach, utilizing responsive HTML/CSS. JavaScript handles real-time updates and API communications, while Chart.js provides visualization capabilities for administrative analytics.

4.7.2 Dual Platform Deployment Strategy

4.7.2.1 Desktop Version

The desktop version serves as the primary monitoring interface, specifically designed for security personnel stationed at monitoring centers or security offices. This version provides comprehensive access to live video feeds from surveillance cameras, enabling real-time visual monitoring of retail spaces.

Key features of the desktop version include:

- **Live Video Feed Access**: Direct integration with surveillance camera systems for real-time monitoring
- **Multi-camera Display**: Simultaneous viewing of multiple camera feeds with grid layout options
- **Enhanced Analytics**: Larger screen real estate allows for detailed statistical displays and comprehensive dashboards
- **Advanced Controls**: Full keyboard and mouse support for efficient navigation and data entry

The desktop interface is optimized for continuous monitoring scenarios, where security personnel can maintain vigilant oversight of multiple store areas simultaneously. The larger display capabilities enable the presentation of detailed alert information, comprehensive statistics, and multi-camera surveillance feeds that would be impractical on mobile devices.

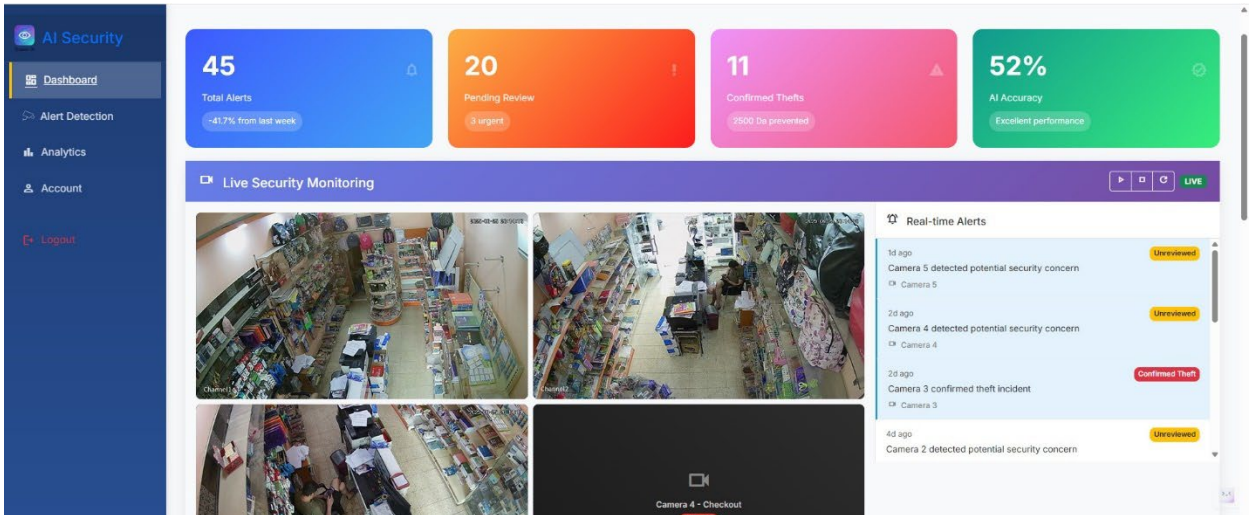

Figure 4-4 Desktop Interface - Multi-Camera Surveillance Dashboard
Source: Author

4.7.2.2 Mobile Version

The mobile version was developed to provide flexibility and mobility for security staff who need to monitor alerts and respond to incidents while moving throughout the retail environment. This version prioritizes alert management and response capabilities over video monitoring due to bandwidth and screen size constraints.

Mobile-specific features include:

- **Touch-Optimized Interface**: Large buttons and touch targets designed for smartphone and tablet interaction
- **Alert-Focused Dashboard**: Streamlined interface emphasizing alert notifications and response actions
- **Offline Capability**: Local data storage for continued operation during network interruptions

The mobile interface utilizes bottom navigation for easy thumb-reach accessibility and implements auto-hide navigation behaviors to maximize screen space utilization. The design philosophy prioritizes immediate alert response capabilities over comprehensive monitoring features.

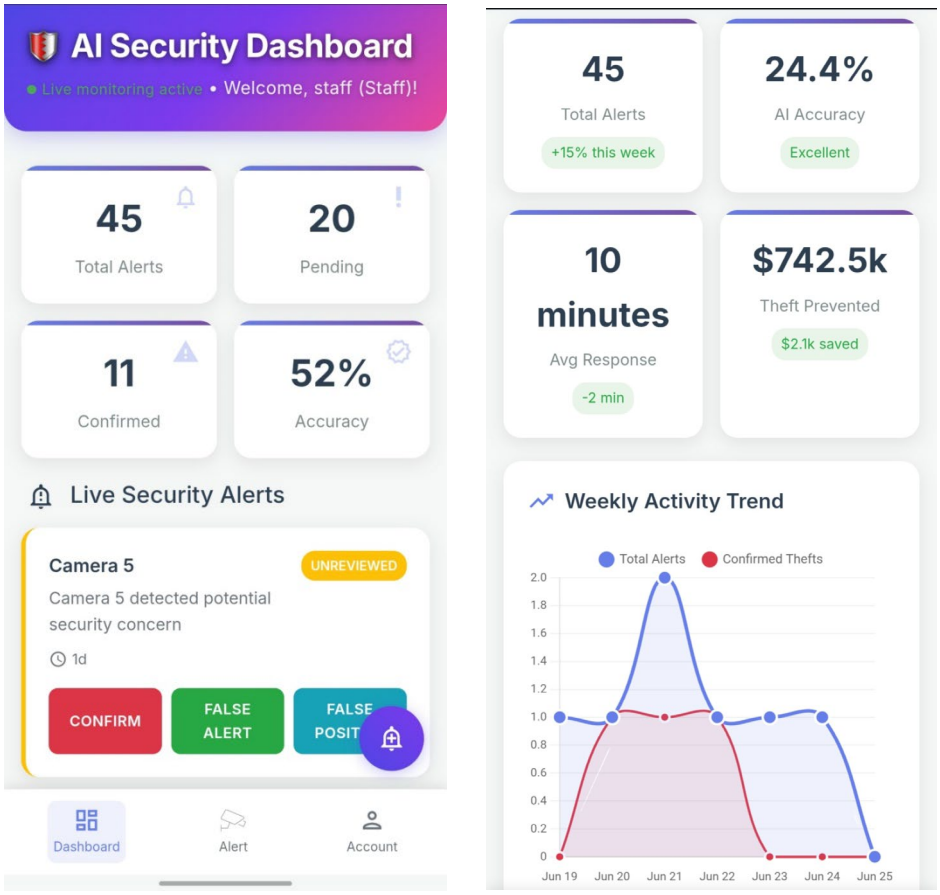

Figure 4-5 Mobile Interface - Alert Management Dashboard
Source: Author

4.7.3 Role-Based Access Control Implementation

4.7.3.1 Administrative Users

Administrative users represent the highest privilege level within the system, typically assigned to security supervisors, store managers, or IT administrators. This role encompasses comprehensive system access and management capabilities.

Administrative privileges include:

- **Complete Analytics Access**: Full access to statistical dashboards, trend analysis, and performance metrics
- **User Management**: Ability to create, modify, and delete user accounts with role assignment capabilities
- **System Configuration**: Access to system settings, alert thresholds, and operational parameters
- **Comprehensive Alert Review**: Access to complete alert history with detailed review and audit trails
- **Report Generation**: Ability to generate comprehensive security reports and analytics exports

Administrators can access all system functionalities across both desktop and mobile platforms, ensuring complete operational oversight and management capabilities.

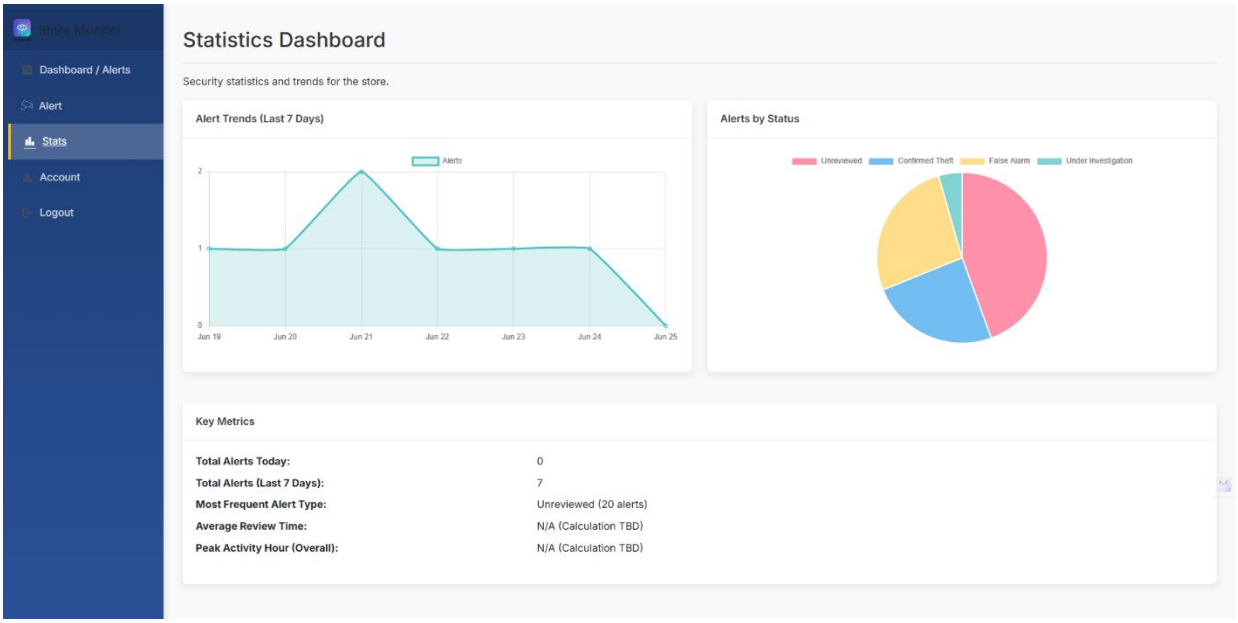

Figure 4-6 Administrator Dashboard - User Management Interface
Source: Author

Figure 4-7 Staff Interface - Alert Response SystemFigure 4-6 Administrator Dashboard - User Management Interface
Source: Author

4.7.3.2 Staff Users

Staff users represent frontline security personnel responsible for immediate alert response and basic monitoring activities. This role is designed for security guards, floor supervisors, and other operational staff members.

Staff user capabilities include:

- **Alert Response**: Primary responsibility for reviewing and acting upon security alerts
- **Status Updates**: Ability to mark alerts as confirmed theft, false alerts, or false positives
- **Comment Addition**: Capability to add contextual comments to alert responses for documentation
- **Basic Monitoring**: Access to real-time alert feeds and immediate response interfaces
- **Limited Analytics**: Access to basic performance metrics relevant to their operational responsibilities

Staff users operate with restricted system access, focusing primarily on alert response and immediate security actions rather than administrative or analytical functions.

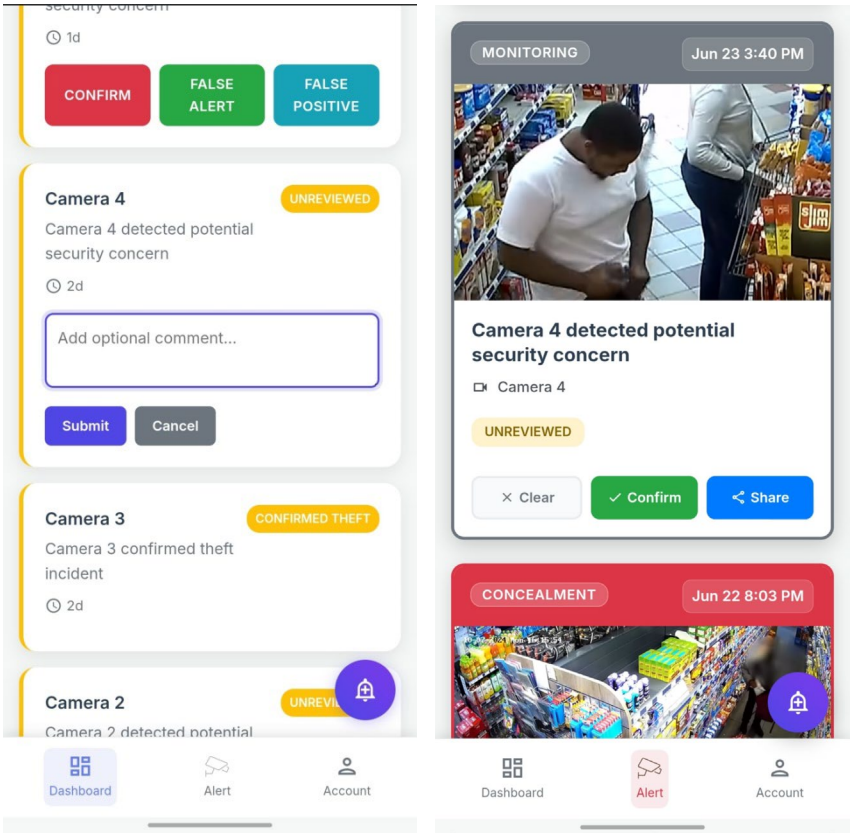

Figure 4-7 Staff Interface - Alert Response System
Source: Author

4.7.4 Key Deployment Considerations

4.7.4.1 Model Optimization for Production

The transition from research prototype to production deployment requires significant optimization of the trained model artifact. The best_shoplifting_detector_model_f1.pth file underwent several optimization processes to ensure efficient real-time performance in retail environments.

Optimization strategies included:

- **Model Quantization**: Reduction of model precision to decrease memory footprint and increase inference speed
- **Hardware Acceleration**: Integration with GPU resources for enhanced processing capabilities
- **Batch Processing**: Implementation of efficient batch processing for multiple camera feeds
- **Memory Management**: Optimization of memory usage patterns for sustained operation

4.7.5 Scalability and Performance

The deployment architecture was designed to accommodate varying scales of retail operations, from single-store implementations to multi-location enterprise deployments. Performance considerations include load balancing, database optimization, and efficient resource management to ensure consistent operation under varying load conditions.

4.7.6 Integration with Existing Systems

The application provides flexible integration capabilities with existing retail security infrastructure, including compatibility with standard surveillance camera protocols and integration with point-of-sale systems for enhanced contextual analysis.

CONCLUSION

This research successfully developed and deployed an innovative dual-stream deep learning system for real-time shoplifting detection in retail environments, addressing a critical challenge that costs the global retail industry over $100 billion annually. The investigation achieved its primary objectives by creating a robust, accurate, and practically deployable solution that significantly advances the state-of-the-art in automated retail security.

5.1 Achievements and Contributions

Technical Innovations The primary innovation lies in the dual-stream architecture that synergistically combines visual context with explicit human motion dynamics. Unlike traditional approaches that analyze entire video frames, our person-centric pose analysis provides targeted behavioral insights. The synchronized data augmentation strategy for both video and pose modalities ensures consistent multimodal learning while addressing class imbalance challenges.

Real-world Applicability and Deployment Insights Successful live deployment demonstrated the system's transition from research prototype to operational application. The comprehensive web-based dashboard with role-based access control accommodates different security personnel needs. Edge computing capabilities enable local processing, ensuring privacy compliance and reducing bandwidth requirements while maintaining real-time performance.

Contribution to Retail Security and AI This work advances automated surveillance from reactive to proactive security paradigms, enabling theft prevention rather than post-incident documentation. The modular architecture and demonstrated scalability provide a foundation for broader retail security applications, while the multimodal approach contributes valuable insights to the computer vision and activity recognition research communities.

5.2 Limitations of the Study

Technical Constraints The system requires fixed camera positions and controlled lighting conditions for optimal performance. The 15-frame sequence requirement may miss very brief suspicious actions, and pose estimation accuracy degrades in crowded environments or with significant occlusions. GPU memory limitations constrain batch processing capabilities and longer temporal sequences.

Data Availability and Quality Issues Limited availability of authentic shoplifting footage necessitated simulated scenarios and synthetic data augmentation. Class imbalance between normal and suspicious behaviors required careful handling through weighted loss functions. The dataset's retail environment diversity was constrained, potentially limiting generalization across different store types and layouts.

Deployment and Scalability Challenges Real-time processing demands substantial computational resources, limiting deployment on standard retail hardware. Integration with legacy surveillance systems requires technical expertise and infrastructure modifications. Scaling to multiple simultaneous camera feeds increases hardware requirements and system complexity.

5.3 Future Work and Research Directions

Enhancements to Detection Accuracy Future improvements include implementing attention mechanisms within the LSTM architecture to focus on critical temporal segments and developing multi-scale temporal analysis for capturing both short-term actions and long-term behavioral patterns. Graph neural networks could better model joint relationships in pose estimation, while adaptive sequence lengths could optimize temporal windows based on specific behaviors.

Expansion to Other Scenarios The framework can be extended to detect additional suspicious activities including vandalism, aggressive behavior, and coordinated group theft. Multi-camera fusion techniques could address occlusion issues and provide comprehensive coverage in large retail spaces. Integration with point-of-sale systems could provide contextual analysis for checkout area monitoring.

Multi-sensor Data Integration Future research should explore incorporating RFID tag monitoring for inventory tracking, thermal sensors for detecting concealed items, and audio analysis for identifying distress signals or suspicious conversations. IoT sensor integration could provide environmental context, while biometric indicators might enhance behavioral understanding.

5.4 Final Thoughts

Reflection on Project Journey and Impact This research successfully demonstrated the transition from academic computer vision research to practical retail security applications. The journey from initial concept through model development, rigorous testing, and real-world deployment highlighted both the potential and challenges of implementing AI in security contexts. The

system's ability to achieve high detection accuracy while maintaining real-time performance represents a significant advancement in automated surveillance technology.

Broader Significance in AI and Security This work contributes to the evolution of intelligent surveillance systems that augment rather than replace human security personnel. The privacy-conscious design and local processing approach addresses growing concerns about surveillance ethics while maintaining security effectiveness. The successful integration of multiple AI technologies—computer vision, pose estimation, and deep learning—demonstrates the power of multimodal approaches in complex real-world applications.

The foundation established by this research creates opportunities for developing comprehensive behavioral analysis systems that could transform security approaches across various domains. As AI technology continues advancing, the principles and methodologies developed here provide a roadmap for creating ethical, effective, and practically deployable security solutions that balance protection needs with privacy rights.

BIBLIOGRAPHY

[1] "Theft rate by country, around the world," TheGlobalEconomy.com. Accessed: Jun. 24, 2025. [Online]. Available: https://www.theglobaleconomy.com/rankings/theft/

[2] "Global Theft Costs $104 Billion Annually | Material Handling and Logistics." Accessed: Jun. 24, 2025. [Online]. Available: https://www.mhlnews.com/global-supply-chain/article/22035119/global-theft-costs-104-billion-annually

[3] "Rising Retail Theft Is a Global, Not American, Problem - Newsweek." Accessed: Jun. 24, 2025. [Online]. Available: https://www.newsweek.com/rising-retail-crime-theft-global-problem-1832450

[4] S. V. Rajenderan and K. F. Thang, "Real-time detection of suspicious human movement," in *international Conference on Electrical Electronics Computer Engineering and their Applications*, 2014.

[5] J. Cheng, X. Zhang, X. Chen, M. Ren, J. Huang, and P. Luo, "Early detection of suspicious behaviors for safe residence from movement trajectory data," *ISPRS Int. J. Geo-Inf.*, vol. 11, no. 9, p. 478, 2022.

[6] R. K. Tripathi, A. S. Jalal, and S. C. Agrawal, "Suspicious human activity recognition: a review," *Artif. Intell. Rev.*, vol. 50, no. 2, pp. 283–339, Aug. 2018, doi: 10.1007/s10462-017-9545-7.

[7] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.

[8] S. Raschka and V. Mirjalili, *Python machine learning: machine learning and deep learning with Python, scikit-learn, and TensorFlow 2*, 3rd ed. Birmingham: Packt Publishing, 2020.

[9] A. Paszke *et al.*, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," 2019, *arXiv*. doi: 10.48550/ARXIV.1912.01703.

[10] Y. Yu, X. Si, C. Hu, and J. Zhang, "A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019, doi: 10.1162/neco_a_01199.

[11] "What is an LSTM Neural Network?," dida Machine Learning. Accessed: Jun. 24, 2025. [Online]. Available: https://dida.do/what-is-an-lstm-neural-network

[12] Pratik, "Implementing LSTM for Human Activity Recognition using Smartphone Accelerometer data," Analytics Vidhya. Accessed: Jun. 24, 2025. [Online]. Available:

https://www.analyticsvidhya.com/blog/2021/07/implementing-lstm-for-human-activity-recognition-using-smartphone-accelerometer-data/

[13] Md. A. Uddin *et al.*, "Deep learning-based human activity recognition using CNN, ConvLSTM, and LRCN," *Int. J. Cogn. Comput. Eng.*, vol. 5, pp. 259–268, Jan. 2024, doi: 10.1016/j.ijcce.2024.06.004.

[14] M. Vakalopoulou, S. Christodoulidis, N. Burgos, O. Colliot, and V. Lepetit, "Fig. 18, [A basic CNN architecture. Classically,...]." Accessed: Jun. 24, 2025. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK597497/figure/ch3.Fig18/

[15] R. Raj and A. Kos, "An improved human activity recognition technique based on convolutional neural network," *Sci. Rep.*, vol. 13, no. 1, p. 22581, Dec. 2023, doi: 10.1038/s41598-023-49739-1.

[16] S. Shinde, A. Kothari, and V. Gupta, "YOLO based Human Action Recognition and Localization," *Procedia Comput. Sci.*, vol. 133, pp. 831–838, Jan. 2018, doi: 10.1016/j.procs.2018.07.112.

[17] "YOLO Object Detection Explained: A Beginner's Guide." Accessed: Jun. 24, 2025. [Online]. Available: https://www.datacamp.com/blog/yolo-object-detection-explained

[18] "What is OpenPose? A Guide for Beginners.," Roboflow Blog. Accessed: Jun. 24, 2025. [Online]. Available: https://blog.roboflow.com/what-is-openpose/

[19] "GitHub - CMU-Perceptual-Computing-Lab/openpose: OpenPose: Real-time multi-person keypoint detection library for body, face, hands, and foot estimation." Accessed: Jun. 24, 2025. [Online]. Available: https://github.com/CMU-Perceptual-Computing-Lab/openpose

[20] M. Grinberg, *Flask web development: developing web applications with Python*, Second edition. Beijing Boston Farnham Sebastopol Tokyo: O'Reilly, 2018.

[21] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016, doi: 10.1109/JIOT.2016.2579198.

[22] G. A. Martínez-Mascorro, J. R. Abreu-Pederzini, J. C. Ortiz-Bayliss, and H. Terashima-Marín, "Suspicious Behavior Detection on Shoplifting Cases for Crime Prevention by Using 3D Convolutional Neural Networks," *Computation*, vol. 9, no. 2, p. 24, Feb. 2021, doi: 10.3390/computation9020024.

[23] J. D. Domingo, J. Gómez-García-Bermejo, and E. Zalama, "Improving Human Activity Recognition Integrating LSTM With Different Data Sources: Features, Object Detection and Skeleton Tracking," *IEEE Access*, vol. 10, pp. 68213–68230, 2022, doi: 10.1109/ACCESS.2022.3186465.

[24] I. Muneer, M. Saddique, Z. Habib, and H. G. Mohamed, “Shoplifting Detection Using Hybrid Neural Network CNN-BiLSMT and Development of Benchmark Dataset,” *Appl. Sci.*, vol. 13, no. 14, Art. no. 14, Jul. 2023, doi: 10.3390/app13148341.

[25] L. Kirichenko, T. Radivilova, B. Sydorenko, and S. Yakovlev, “Detection of Shoplifting on Video Using a Hybrid Network,” *Computation*, vol. 10, no. 11, Art. no. 11, Nov. 2022, doi: 10.3390/computation10110199.

[26] “CRCV | Center for Research in Computer Vision at the University of Central Florida.” Accessed: Jun. 24, 2025. [Online]. Available: https://www.crcv.ucf.edu/projects/real-world/

ANNEX

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Abou Bekr Belkaid Tlemcen

Business Model Canvas

BMC

N° de projet : FS - 065

Faculté/Institut : Faculté des Sciences

Département : Informatique

Nom du projet :

Détection de comportements suspects et prévention des vols via une solution d'intelligence artificielle en temps réel.

Encadrant: Mme Benosman Amina

Etudiants: Bedjeboudja Anas

Baseer AI

Année universitaire : 2024/2025

1- Proposition de valeur (Value Proposition)

a. Quels problèmes résolvons-nous pour nos clients ?

Nous résolvons le problème du vol à l'étalage, qui engendre des pertes financières significatives pour les commerçants. Nous adressons également le manque de solutions de surveillance proactives et en temps réel, la difficulté pour le personnel de surveiller efficacement plusieurs flux de caméras simultanément, et le besoin d'un système de sécurité qui fonctionne sans connexion Internet constante, une problématique importante en Algérie où la connectivité peut être un enjeu. Enfin, nous comblons un vide sur le marché algérien où une telle technologie n'est pas encore disponible.

b. Quels besoins de nos clients satisfont nos produits ou services ?

Nos produits satisfont le besoin crucial de sécurité renforcée et de prévention des pertes dans les points de vente. Ils répondent au besoin d'être alerté immédiatement en cas d'activité suspecte, permettant une intervention rapide. Nos dispositifs valorisent l'infrastructure de surveillance existante (caméras) en y ajoutant une couche d'intelligence. Ils comblent le besoin d'une solution autonome, particulièrement adaptée aux environnements avec une connectivité Internet limitée ou inexistante, et offrent une technologie de sécurité moderne et efficace.

c. En quoi notre offre est-elle différente de celle de nos concurrents ?

Notre offre est unique sur le marché algérien : nous sommes les premiers à proposer un système de détection de vol à l'étalage basé sur l'IA et l'informatique en périphérie (edge computing). Contrairement à des solutions logicielles qui pourraient exister ailleurs, nous fournissons un dispositif matériel dédié qui s'intègre facilement et fonctionne sans dépendance à Internet pour sa fonction principale. Les alertes en temps réel sur une application mobile, ainsi que la scalabilité de nos solutions (appareils pour 4, 8 ou 16 caméras), nous distinguent également.

d. Quelle est notre proposition unique de valeur ?

Nous offrons la première solution de détection de vol à l'étalage par Intelligence Artificielle en Algérie : un dispositif matériel innovant, fonctionnant en périphérie (edge computing) sans nécessiter de connexion Internet, qui s'intègre de manière transparente aux systèmes de caméras existants pour envoyer des alertes en temps réel au personnel via une application mobile, assurant ainsi une sécurité proactive et une réduction des pertes.

2- Segments de clients (Customer Segments)

a. Quels sont nos clients principaux?

Nos clients principaux sont les propriétaires et gérants de commerces de détail en Algérie, quelle que soit leur taille.

b. Quels sont les différents segments de clients que nous visons ? Nous visons plusieurs segments :

- Les petits commerces de détail (boutiques, supérettes, pharmacies).
- Les moyennes surfaces (supermarchés, magasins d'électronique, magasins de vêtements).
- Les grandes surfaces de vente (hypermarchés, grands magasins).
- Potentiellement, d'autres types d'entreprises confrontées à des risques de vol interne ou externe (entrepôts, zones de stockage).

c. Quels sont les besoins spécifiques de chaque segment de clients?

- Petits commerces : Solution abordable, facile à installer et à utiliser, adaptée à un nombre limité de caméras, et offrant un retour sur investissement rapide par la réduction des vols.
- Moyennes surfaces : Système capable de gérer un plus grand nombre de caméras, un système d'alerte robuste pour informer plusieurs membres du personnel, et une intégration simple.
- Grandes surfaces : Solution hautement performante pour un parc de caméras étendu, capable de s'intégrer avec les équipes de sécurité et protocoles existants, et potentiellement des fonctionnalités d'analyse plus poussées.

d. Comment pouvons-nous catégoriser nos clients en groupes distincts? Nous pouvons catégoriser nos clients en fonction de :

- La taille du commerce/nombre de caméras : Ceux nécessitant un appareil pour 4 caméras, pour 8 caméras, ou pour 16 caméras.
- Le type de commerce : Alimentaire, électronique, habillement, pharmacie, etc., car les types de vol et les points sensibles peuvent varier.
- Le niveau de maturité technologique ou le budget sécurité.

3- Relation avec les clients (Customer Relationships)

a. Quel type de relation chaque segment de clients attend-il de nous ?

Tous les segments attendent un produit fiable et performant. Ils s'attendent à un support technique réactif pour l'installation et en cas de problème. Une communication claire sur les fonctionnalités du produit est également attendue. Les clients plus importants (moyennes et grandes surfaces) pourraient attendre un accompagnement plus personnalisé et un gestionnaire de compte dédié.

b. Comment entretenons-nous actuellement les relations avec nos clients ?

Nous prévoyons d'entretenir les relations par :

- Des ventes directes et personnalisées.
- Un support à l'installation (sur site ou à distance).
- Un service après-vente accessible (téléphone, email).
- Potentiellement, des notifications via l'application mobile pour des mises à jour mineures de l'application elle-même (celles ne touchant pas au cœur du dispositif qui est hors ligne).

c. Comment pouvons-nous améliorer ou personnaliser nos interactions avec nos clients ?

Nous pouvons améliorer nos interactions en :

- Proposant des services d'installation et de formation clairs et efficaces.
- Fournissant une documentation utilisateur complète et accessible (en français et en arabe).
- Mettant en place un suivi post-installation pour s'assurer de la satisfaction client.
- Collectant activement les retours clients pour améliorer nos produits.
- Offrant des niveaux de support différenciés en fonction des besoins et de la taille du client.

4- Canaux de distribution (Channels)

a. Par quels canaux nos clients veulent-ils être atteints ?

Nos clients veulent être atteints par :

- Une équipe de vente directe qui peut leur présenter la solution et comprendre leurs besoins spécifiques.
- Une présence en ligne (site web informatif, réseaux sociaux professionnels) pour découvrir le produit et prendre contact.
- Potentiellement, par des installateurs de systèmes de sécurité qu'ils connaissent et en qui ils ont confiance.

b. Quels canaux sont les plus efficaces pour atteindre chaque segment de clients ?

- Petits commerces : Prospection directe, marketing digital ciblé (publicités locales en ligne), bouche-à-oreille.
- Moyennes et grandes surfaces : Approche commerciale directe et structurée, présentations personnalisées, participation à des salons professionnels (si pertinent en Algérie), partenariats avec des intégrateurs de solutions de sécurité.

c. Comment pouvons-nous intégrer différents canaux pour améliorer l'expérience client ?
Nous pouvons intégrer nos canaux en assurant une communication cohérente :

- Notre site web sert de vitrine et de point d'information initial.
- Notre équipe commerciale prend le relais pour des démonstrations et des conseils personnalisés.
- L'application mobile est au cœur de l'expérience utilisateur pour les alertes et pourrait intégrer une section d'aide ou de contact support.
- Le support client par téléphone et email assure la continuité du service.

5- Partenaires clés (Key Partnerships)

a. Qui sont nos partenaires clés ? Nos partenaires clés pourraient inclure :
   - Fournisseurs de composants électroniques pour la fabrication de nos dispositifs.
   - Fabricants ou distributeurs de systèmes de caméras (pour assurer la compatibilité et explorer des offres groupées).

- Installateurs de systèmes de sécurité et intégrateurs (qui pourraient agir comme revendeurs ou partenaires d'installation).
- Potentiellement, des entreprises de logistique pour la distribution des appareils.

b. Quels sont les partenariats qui nous aident à réduire les coûts, à accéder à de nouvelles ressources ou à améliorer notre proposition de valeur ?
   - Fournisseurs : Des partenariats solides peuvent mener à des prix d'achat de composants plus avantageux et à une chaîne d'approvisionnement fiable.
   - Installateurs/Intégrateurs : Ils peuvent étendre notre portée sur le marché sans avoir à développer une large équipe d'installation interne, et apporter leur expertise locale.
   - Distributeurs de caméras : Assurer la compatibilité technique et potentiellement des actions de co-marketing.

c. Comment pouvons-nous aligner nos intérêts avec ceux de nos partenaires ?

Nous pouvons aligner nos intérêts par :

- Des accords clairs et mutuellement bénéfiques (marges pour les revendeurs, conditions d'achat pour les fournisseurs).
- Une communication transparente et régulière.
- Le partage d'informations pertinentes (retours clients, évolutions techniques).
- Des initiatives marketing conjointes.

6- Activités clés (Key Activities)

a. Quelles sont les actions principales que nous devons entreprendre pour livrer notre proposition de valeur ?

Nos actions principales sont :

- La recherche et développement continus pour l'algorithme d'IA et le matériel.
- La fabrication ou l'assemblage des dispositifs de détection.
- Le développement et la maintenance du logiciel embarqué et de l'application mobile.
- Les activités de marketing et de vente pour atteindre nos clients cibles.
- L'installation des dispositifs et le support technique client.

b. Quelles sont les opérations essentielles pour notre entreprise ?

Les opérations essentielles incluent :

- La production et le contrôle qualité des dispositifs.
- La gestion de la chaîne d'approvisionnement.
- Le développement technologique (IA, software, hardware).
- Le service client et le support technique.
- La gestion commerciale et administrative.

c. Quelles sont les activités qui créent le plus de valeur pour nos clients ?

Les activités qui créent le plus de valeur sont :

- La précision et la fiabilité de la détection de vol par l'IA.
- La facilité d'intégration de notre dispositif avec les systèmes de caméras existants.
- La notification en temps réel des incidents.
- L'autonomie du système (fonctionnement sans Internet).
- Un support client efficace et réactif.

7- Ressources clés (Key Resources)

a. Quels sont nos actifs matériels, immatériels et humains essentiels ?
   - Actifs matériels : Les composants pour fabriquer les dispositifs, l'équipement d'assemblage et de test, les locaux (bureau, atelier).
   - Actifs immatériels : L'algorithme d'IA propriétaire et le logiciel associé, la marque, les brevets potentiels, les données d'entraînement de l'IA (anonymisées et sécurisées), notre expertise technique.
   - Actifs humains : Ingénieurs en IA et en logiciels, ingénieurs hardware, techniciens d'assemblage, équipe commerciale et marketing, personnel de support technique.

b. Quels sont les outils, les technologies ou les partenariats dont nous avons besoin pour réussir ?
   - Outils et technologies : Plateformes de développement IA, outils de développement logiciel (pour l'embarqué et le mobile), outils de conception assistée par ordinateur (CAO) pour le matériel, technologies d'informatique en périphérie (edge computing).
   - Partenariats : Fournisseurs fiables de composants, potentiellement des partenaires de distribution ou d'installation.

c. Quels sont les principaux avantages concurrentiels de nos ressources ? Nos principaux avantages concurrentiels liés à nos ressources sont :
   - Notre algorithme d'IA spécialisé et entraîné pour la détection de vol.
   - Notre expertise dans le déploiement de solutions d'IA en edge computing.
   - L'avantage d'être pionnier sur le marché algérien avec cette offre spécifique.
   - La capacité à offrir une solution ne dépendant pas d'Internet, répondant à un besoin local fort.

8- Charges et coûts (Cost Structure)

a. Quels sont les coûts fixes et variables associés à notre modèle économique ?
   - Coûts fixes : Salaires (R&D, administration, support de base), loyer des locaux, amortissement du matériel de production et de R&D, licences logicielles de développement, assurances.
   - Coûts variables : Coût des composants par appareil produit (COGS), coûts d'assemblage par unité, commissions sur les ventes, frais de marketing et de publicité par campagne, frais de transport et de logistique.

b. Quels sont les coûts les plus importants pour notre entreprise ? Les coûts les plus importants seront probablement :
   - La recherche et développement (surtout l'investissement initial dans l'IA et le prototypage) 120 000 - 300 000 Da / mois
   - Le coût des matières premières (composants électroniques) 45 000 - 120 000 Da / unité
   - Les salaires du personnel qualifié (ingénieurs IA, développeurs) 10 000 - 150 000 Da / mois

- Les dépenses de marketing et de vente pour pénétrer le marché et éduquer les clients. 10 000 - 30 000 Da / mois

c. Comment pouvons-nous réduire les coûts ou améliorer l'efficacité de nos opérations ? Nous pouvons chercher à :
   - Optimiser notre chaîne d'approvisionnement (négocier les prix avec les fournisseurs, commandes en volume).
   - Améliorer l'efficacité du processus d'assemblage et de test.
   - Standardiser nos plateformes logicielles et matérielles autant que possible.
   - Mettre en place des processus de support client efficaces pour réduire le temps de résolution par incident.
   - Cibler nos efforts marketing pour maximiser le retour sur investissement.

9- Revenus (Revenue Streams)

a. Quels produits ou services nos clients sont-ils prêts à payer ?

Nos clients sont prêts à payer pour :

- Le dispositif de détection de vol lui-même (adapté pour 4, 8 ou 16 caméras).
- Potentiellement, des frais d'installation si celle-ci est complexe ou si le client souhaite un service clé en main.
- Optionnellement à l'avenir : des services de maintenance étendue ou des garanties prolongées.

b. Quels sont les différents moyens par lesquels nous pouvons générer des revenus ?

Nous générons des revenus principalement par :

- La vente unique de nos dispositifs matériels. Chaque vente d'un appareil (pour 4, 8 ou 16 caméras) constitue une source de revenu.

c. Quel est notre modèle de tarification ?

Notre modèle de tarification est basé sur :

- Un prix fixe par type d'appareil (un prix pour le modèle 4 caméras, un autre pour le 8 caméras, et un troisième pour le 16 caméras).
- Vente directe du dispositif :
  • Modèle pour 4 caméras : 80 000 Da
  • Modèle pour 8 caméras : 130 000 Da
  • Modèle pour 16 caméras : 200 000 Da
- Il s'agit d'un paiement unique à l'achat du matériel, qui inclut le logiciel d'IA embarqué.
- Aucun frais d'abonnement récurrent pour la fonctionnalité de base, puisque le système fonctionne sans Internet.

Business Model Canvas

جامعة أبو بكر بلقايد
كلية التكنولوجيا
UNIVERSITÉ DE TLEMCEN

Partenaires clés

- Distributeurs de caméras de sécurité
- Installateurs/intégrateurs de systèmes de sécurité
- Partenaires logistiques

Activités clés

- Recherche et développement IA & hardware
- Développement et maintenance logiciel (embarqué & mobile)
- Marketing et vente
- Support client et installation

Ressources clés

- Algorithme IA propriétaire
- Données d'entraînement
- Dispositif edge (matériel)
- Équipe technique (IA, dev, hardware)

Proposition de valeur

Bassir AI propose une solution intelligente, autonome et locale de prévention du vol à l'étalage, qui :

- Réduction des pertes grâce à une détection automatique des comportements suspects.
- Alertes en temps réel
- Fonctionnement sans Internet
- Intégration simple avec le system de surveillance existant
- Première solution IA de ce type en Algérie

Relation clients

- Vente directe et accompagnement à l'installation
- Documentation claire (FR & AR)
- Support client
- Suivi après-vente
- Recueil de feedback

Canaux de distribution

- Équipe de vente directe
- Site web professionnel
- Réseaux sociaux & marketing digital (sponsor)
- Partenariat avec installateurs locaux
- Participation à des salons ou événements tech

Segments clients

- Petits commerces (petit magasin, comitique, pharmacies, etc)
- Moyennes surfaces (supérettes, supermarchés, magasins spécialisés)
- Grandes surfaces (hypermarchés, entrepôts)
- Institutions sensibles (écoles, hôpitaux, etc.)

Structure de coûts

Composants électroniques (par dispositif) 45 000-120 000 Da / unité

Assemblage et embalage (main d'œuvre) 1000-2000 Da / unité

Marketing Réseaux sociaux campagne locale 10 000 - 30 000 Da / mois

Sources de revenus

Vente directe du dispositif :

- Modèle pour 4 caméras : 80 000 Da
- Modèle pour 8 caméras : 130 000 Da
- Modèle pour 16 caméras : 200 000 Da