Deep Feature-based Anomaly Detection for Video Surveillance

A Research study submitted in fulfilment of the requirements for the degree of

Masters in Computer Science

in the

Department of Computer Science

Lakehead Univeristy,

955 Oliver Road,

Thunder Bay, ON , P7B 5E1,

Canada

April 2023

Authors:

Dhruva Shah

Parth Bhatt

Supervisor:

Dr. Thangarajah Akilan

DECLARATION

We hereby declare that the dissertation incorporates material that is the result of joint research, as follows: The dissertation also incorporates the outcome of research under the supervision of Dr Thangarajah Akilan and collaboration with Dhruva Shah and Parth Bhatt. In all cases, the Introduction, Literature review, Proposed Method, Implementation, and Results were performed by the authors, and the contribution provision of proofreading and reviewing the research papers regarding technical content. We are aware of Lakehead University's policy on Authorship and,We certify that We have properly acknowledged the contribution of other researchers to my dissertation, have obtained written permission, and have obtained written from each of the co-authors. We certify that, with the above qualification, this dissertation, and the research to which it refers, is the product of our own work.

ACKNOWLEDGEMENT

Our team would like to sincerely thank Lakehead University in Thunder Bay, Ontario, for providing us with excellent educational assistance and services. We would like to express our gratitude to Lakehead University's faculty and staff for their support throughout the academic year.

We appreciate our parents and family's unwavering support, advice, and prayers. They have given us a lot of support, enthusiasm, effort, and priceless assistance in order to help us realise our goals.

We would like to extend our sincere gratitude to Dr. Thangarajah Akilan, our supervisor, for his assistance with our research study and for serving as a beacon of guidance throughout this research study. We appreciate your tireless efforts in assisting us in completing the research project. We would also want to express our gratitude for the important contribution made by our fellow classmates in formulating insightful questions concerning our research work that improved our performance.

Lakehead
UNIVERSITY

Contents

List of Figures

List of Tables

ABSTRACT

Detecting anomalies in video surveillance is a challenging task that requires distinguishing between normal and abnormal behavior. Video surveillance systems not only face challenges in identifying and monitoring unusual human actions but also in differentiating normal from anomalous actions due to a large amount of data in video format. In this study, a deep feature-based anomaly detection approach is proposed that utilizes a two-stream deep learning model with an Inflated 3D Network (I3D) as the feature extraction component.

The UCF Crime dataset [6], which contains various crime scenarios, is used to evaluate the performance of the proposed method. The I3D two-stream architecture is a state-of-the-art model that combines spatial and temporal information from videos for feature extraction. In this work, we extract features from the UCF Crime dataset using the I3D two-stream architecture and train a binary classifier to detect anomalies. Our approach achieves a promising average detection accuracy of 87.52%, demonstrating the effectiveness of the I3D two-stream architecture for video anomaly detection. Overall, our study provides a valuable contribution to the field of video anomaly detection and demonstrates the potential of deep learning models for improving video surveillance systems.

Notations and Abbreviations

.

DL:	Deep Learning
UCF:	University of Central Florida
FCNN:	Fully Connected Neural Network
I3D:	Inflated 3D ConvNet
C3D:	Convolutional 3D
MIL:	Muliple Instance Learning
ReLU:	Rectified Linear Unit
AE:	Autoencoder
CAE:	Convolutional Autoencoder
CNN:	Convolutional Neural Network
MGFN:	Magnitude-Contrastive Glance-and-Focus Network
RTFM:	Robust Temporal Feature Magnitude Learning

Chapter 1

Introduction

1.1 Overview

Anomaly detection in video is a long-standing issue in computer vision, with several applications in surveillance monitoring, such as identifying illicit activities, traffic accidents, and strange incidents, among others. Millions of surveillance cameras are being installed in public locations across the world. Nevertheless, the majority of the cameras are only passively recording and do not have any monitoring capacity. With large amounts of data created by video cameras every minute, human effort is insufficient to comprehend this massive corpus of video data. We require machine vision to detect irregularities in video automatically. Detecting anomalies in the video is a challenging problem due to the ambiguous nature of what constitutes an anomaly. Any event that deviates from normal behavior can be classified as an anomaly, making it impossible to collect training data that encompasses all possible abnormal events. This makes it challenging to solve this problem using a standard classification framework. According to a report by MarketsandMarkets [1], the video analytics market is expected to grow from USD 3.23 billion in 2018 to USD 8.55 billion by 2023, at a Compound Annual Growth Rate (CAGR) of 21.5% during the forecast period. The report highlights the increasing need for advanced video surveillance and security solutions in various applications, including video anomaly detection.

Classifying data as normal or abnormal based on specific patterns is an important activity called anomaly detection. Three types of anomaly detection that are based on the annotated training data and the chosen algorithms can accomplish this task. Unsupervised anomaly detection, which is the first type, makes the assumption that movies with hidden anomalies have significant reconstruction flaws. Due to a lack of understanding of anomalies in anomalous videos and an inability to pick up on the typical patterns in typical videos, this technique frequently performs badly. The best performance is anticipated from supervised anomaly detection, the second type of anomaly detection. This method necessitates the annotation of individual video frames, which can be laborious and error-prone. Despite having the potential for great accuracy, supervised anomaly detection is not well-studied since it necessitates annotation.

Weakly supervised video anomaly detection, the third type of anomaly detection, just needs annotation at the video level. This method is simpler to use and less susceptible to human mistakes. It has thus drawn increasing interest in the field of video anomaly detection. Multiple anomalies in a single video and an imbalance between normal and anomalous movies are just two of the difficulties that weakly

[1] https://www.marketsandmarkets.com/Market-Reports/intelligent-video-analytics-market-778.html

supervised anomaly detection algorithms face because they rely on learning from a small number of labels. Researchers have created a variety of algorithms, including auto-encoders, one-class SVMs, and GANs, among others, to address these issues.

The objective of this research is to enhance the detection of anomalies in video surveillance by implementing a deep feature-based methodology. Our approach uses a two-stream deep learning model with Inflated 3D Convolutional Neural Networks (I3D) [5] as the feature extraction part and the UCF Crime dataset for training. Specifically, the I3D model is pre-trained on large-scale action recognition datasets and fine-tuned on the UCF Crime dataset, which contains normal and abnormal activities. We use the Multiple Instance Learning (MIL) loss function [6] with a threshold to classify video segments as normal or abnormal. Our proposed method achieves an Area Under the Curve (AUC) of 87.52% and demonstrates the effectiveness of the I3D two-stream architecture for video anomaly detection. We also visualize the output using a heat map and a graphical representation of the temporal analysis of anomaly prediction on videos. Our experimental results show a significant improvement in anomaly detection performance compared to state-of-the-art methods. Our abstract level flow diagram is shown in Fig. 1.1

1.2 Problem Statement and Motivation

With the increasing prevalence of crime in our surroundings, it has become imperative to implement a smart surveillance system that can detect and alert authorities about any anomalies in the video stream in real time. Prompt reaction from relevant authorities is essential to minimize the loss of life and property. The smart surveil-

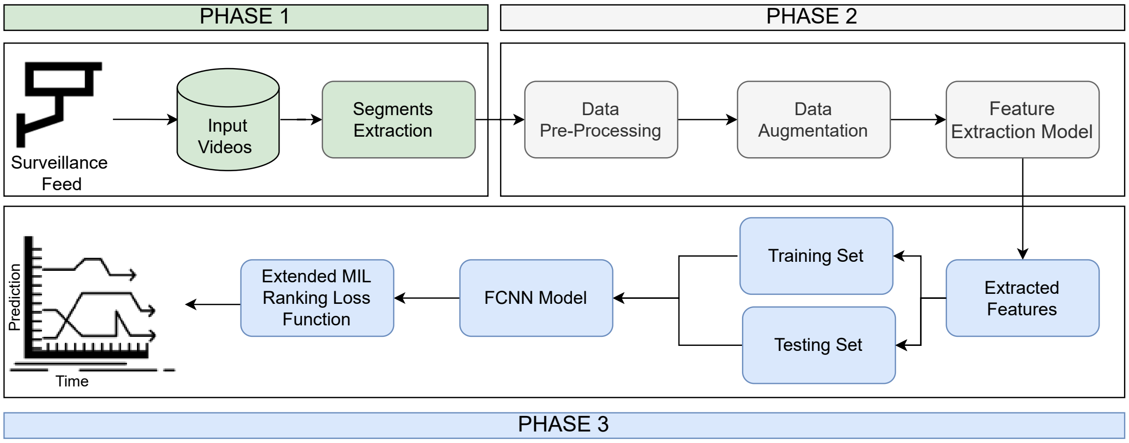


Figure 1.1: Complete Flow Diagram of Proposed Solution. It consists of three phases, **Phase 1**: Input, **Phase 2**: Feature Extraction and **Phase 3**: Classification.

lance system should be equipped with advanced algorithms for detecting anomalies in the video stream. The system should be able to identify and flag any unusual occurrences, including equipment malfunctions and process errors, to prevent failures or accidents.

Anomaly detection is a critical component of modern surveillance systems as it can provide early warning of potential threats and enable timely intervention by relevant authorities. Moreover, it can help in improving the overall efficiency of surveillance systems by reducing false alarms and minimizing response times. Anomaly detection algorithms can be trained using machine learning techniques to improve their accuracy and reduce false positives.

The development of more advanced and efficient surveillance and security systems can enhance public safety, security, and efficiency in various domains. The use of smart surveillance systems with advanced anomaly detection algorithms can significantly reduce the incidence of crime and help in maintaining law and order. Thus, there is a need for continued research and development in this field to improve the effectiveness and efficiency of smart surveillance systems.

1.3 Challenges

Video anomaly detection in deep learning is a challenging task due to various technical challenges. Some of the major challenges are:

1. **Ambiguousness**: Anomaly detection is generally understood as identifying events that are not supposed to occur in a particular context. However, in real-world scenarios, it can be difficult to clearly distinguish between normal and abnormal occurrences. Some normal occurrences may exhibit unusual features similar to abnormal events, which can make it challenging to accurately detect anomalies using models.

2. **Data Quality and Quantity**: Deep learning models require a large amount of high-quality training data to perform accurately. Obtaining high-quality training data is difficult, and annotated data is scarce. In the case of video anomaly detection, labeled video datasets are even rarer, making it difficult to train deep learning models effectively.

3. **Model Selection**: Choosing the right deep-learning model for video anomaly detection is a critical challenge. Various deep learning models are available, each with its strengths and weaknesses. It is essential to choose the right model that can detect anomalies accurately while remaining computationally efficient.

4. **Computational Complexity**: Video anomaly detection requires processing large amounts of data, which can be computationally expensive. The high computational complexity of deep learning models can lead to slower training times, making it challenging to train large models.

5. **Real-time Processing**: Real-time video anomaly detection is crucial in many applications, such as surveillance and security. However, deep learning models require significant processing power, making it challenging to perform real-time processing of video feeds.

6. **Interpreting Model Outputs**: Deep learning models for video anomaly detection are often considered black boxes, meaning it is challenging to interpret how the model is making predictions. Understanding how the model arrived at its output is crucial for improving model performance and building trust in the system.

7. **Robustness**: Deep learning models are often trained on ideal conditions and can be susceptible to overfitting. It is essential to ensure that the model is robust enough to handle various environmental conditions, such as lighting changes and occlusions.

8. **Generalization**: Deep learning models for video anomaly detection must be generalizable to new environments and scenarios. It is challenging to ensure that the model can perform accurately in different situations without the need for significant retraining.

9. **Privacy Concerns**: Video anomaly detection systems raise privacy concerns as they collect and process sensitive data. It is crucial to ensure that such

systems are designed with privacy in mind, such as employing techniques such as differential privacy to protect individuals' privacy.

10. **Noise**: As video surveillance has become more prevalent in various public and private settings, cameras are being used to enhance safety, and they can be found in locations such as elevators, crossroads, shopping malls, restaurants, and even homes. Although it is easy to obtain video surveillance data using existing imaging equipment, manually labeling the data is a time-consuming task and can result in errors. The presence of noise in the data is likely to affect the accuracy of models used for anomaly detection in the long run.

To overcome these challenges, innovative approaches such as data augmentation, transfer learning, and advanced model architectures are required to develop accurate and robust video anomaly detection models that can generalize to novel scenarios and environments

1.4 Objectives and scope

Video anomaly detection in deep learning aims to identify anomalous events or behavior in video data to improve surveillance systems, enhance public safety, and automated anomaly detection. The scope of these systems involves analyzing video data from various sources and utilizing machine learning algorithms such as CNNs and RNNs to detect anomalies in real-time. In addition, these systems may also analyze other types of data such as audio or sensor data to improve accuracy. The objectives of video anomaly detection in deep learning include detecting anomalies, improving surveillance systems, automating anomaly detection, and improving data analysis. Overall, these systems have significant potential applications and can help prevent crime and improve public safety by detecting suspicious behavior or events in real-time.

1.5 Contributions

Anomaly detection in videos has been researched for more than a decade, and because of its broad applicability, this field has attracted the attention of many researchers. Its main application is to improve the safety of public lives and assets, through video surveillance systems, which are widely used in public places such as markets, shopping malls, hospitals, banks, streets, educational institutions, city administrative offices, and smart cities. Most of the time, the primary goal of security applications is the timely and accurate detection of video anomalies. As a result, many approaches have been proposed over the years, ranging from statistical to machine learning. Although several comprehensive reviews of deep learning-based anomaly detection can be found in the literature [7], [8], [9], [10], [11], [12], [13], [14], [15] and [16].Suarez *et al.* [7], it has provided an overview of recent advances in video anomaly detection using deep learning techniques. In terms of the final step in identifying anomalies, four types of approaches have been introduced: using reconstruction error, predicting future frames, using classification, and using scoring. It has also presented the various commonly used datasets, along with key details such as video resolution and examples of anomalies discovered within the respective datasets. Chalapathy *et al.* [8], look into and identify various deep learning models for anomaly detection, as well as assess their suitability for various datasets. When applying a deep learning model to a specific domain or set of data, these assumptions can be used as guidelines to evaluate the technique's effectiveness in that domain. Nayak *et al.* [9], the existing deep-learning methods for detecting video anomalies have been classified into several groups based on modeling techniques. Following that, a comparative analysis of existing deep learning-based video anomaly detection methods is provided to aid in the selection of a specific method that works best for a specific application. Furthermore, an in-depth examination of performance evaluation methodologies in terms of datasets, computational infrastructure, evaluation criteria, and performance metrics for quantitative and qualitative analyses is provided. Finally, the current and open research challenges in deep learning approaches for video anomaly detection are briefly discussed. In Munyua *et al.* [10], deep learning models were categorized based on their learning techniques and design architecture. The mechanism of anomaly detection was also discussed in the papers, along with the most popular datasets for training and testing models, as well as the evaluation criteria used in the reviewed works. Zhu *et al.* [11], present the different approaches of the three main categories, i.e. unsupervised, weakly supervised, and supervised, based on the experimental settings on the training data. It also discusses benchmark datasets and open problems pertaining to video anomaly detection in smart surveillance systems. In Kiran *et al.* [12], they review the state-of-the-art deep learning-based methods for video anomaly detection and categorize them based on the type of model and criteria of detection. It also performs simple studies to understand the different approaches and provides the criteria of evaluation for spatiotemporal anomaly detection. Ramachandra *et al.* [13], summarises research trends in the field of anomaly detection in single-scene video feeds. It organizes and categorizes previous research into an easy-to-understand taxonomy and provides a comprehensive comparison of the accuracy of many algorithms on standard test sets, also talks about different problem formulations, publicly available datasets, and evaluation criteria. In the end, it provides best practices and suggests some future research directions.Geetha *et al.* [14], it presents a comprehensive analysis of machine vision-based fire/smoke detection methods and their performance. Firstly, it discusses the fundamentals of image processing methods, CNNs, and their potential applications in video smoke and fire detection. Also, it sheds light on the existing datasets and provides a summary of recent methodologies in this field. Finally, it talks about the challenges and improvements needed to further the development of CNN applications in this field. Anoopa *et al.* [15], present a comparative study of different anomaly detection methods in deep learning and representation learning and the limitations of each method. In addition, it also outlines open research challenges for future research. Yadav *et al.* [16], aims to provide a comprehensive examination of video anomaly detection systems that use deep learning that has been published since 2019. It also investigates the learning methods and the accuracy of the model on the training and testing datasets. Furthermore, it provides a brief overview of video datasets for anomaly detection and emerging trends in this topic. However, none of them organized the survey on the basis of the Learning technique and state-of-the-art performance models like Graph neural networks and Transformers. To fill this gap, we aim to provide a comprehensive review of deep learning techniques for video anomaly detection in this paper, including the problem formulation, hierarchical categorization, limitations, and performance analysis on various datasets.

We examine published articles, conference papers, and high-quality preprints (i.e., arXiv) on deep learning techniques for video anomaly detection from 2016 to the present. We have tried our best to cover all the recent studies, but we may, however, overlook some recently published and pre-published studies, which is unavoidable. In summary, the primary contributions of this literature review are as follows:

- A Comprehensive Review of the Deep Learning Techniques for Video Anomaly Detection Based on Different Deep Learning Techniques and SOTA Models Like GNN and Transformers.
- We have studied most of the relevant studies on this topic and present a detailed overview of them along with their limitations and performance on various datasets.

1.6 Organization

The content is organized into four main sections: Introduction, Related Work, Proposed Method, and Experimental Analysis. The Introduction section provides an overview of the problem statement and motivation, challenges, objectives, and contributions of the research. The Related Work section presents the background information and a literature review on transfer learning-based models, ensemble learning-based models, continual learning-based models, reinforcement learning-based models, representation learning-based models, generative models, and GNN-based models. Additionally, the section also describes benchmark datasets. The Proposed Method section outlines the methodology of the research and describes the dataset, input, data preprocessing, feature extraction, model architecture, model training, and hyperparameters. The Experimental Analysis section provides the results of the experimental study and discusses the conclusion and future direction of the research. Overall, the content is well-organized and structured, providing a clear understanding of the research methodology and its outcomes.

Table 1.1: Comparison of existing surveys on Video Anomaly Detection

Article	**TL**	**EL**	**CL**	**RL**	**REL**	**GM**	**GNN**	**Benchmark Datasets**
Suarez *et al.* [7]	✓	✓	✓	✓	✓	✓	-	-
Chalapathy *et al.* [8]	✓	✓	✓	✓	-	✓	-	-
Nayak *et al.* [9]	✓	✓	✓	✓	✓	✓	-	-
Munyua *et al.* [10]	✓	✓	✓	✓	-	✓	-	-
Zhu *et al.* [11]	✓	✓	-	✓	-	✓	-	-
Kiran *et al.* [12]	✓	-	-	-	✓	✓	-	-
Ramachandra *et al.* [13]	✓	✓	✓	-	-	✓	-	-
Geetha *et al.* [14]	✓	✓	-	-	-	✓	-	-
Anoopa *et al.* [15]	✓	✓	-	-	✓	✓	-	-
Yadav *et al.* [16]	✓	✓	-	-	✓	✓	-	-
Ours	✓	✓	✓	✓	✓	✓	✓	✓

”-” indicates surveys do not include those parameters. TL = Transfer Learning based Models, EL = Ensemble Learning based Models, CL = Continual Learning based Models, RL = Reinforcement Learning based Models, REL = Representation Learning based Models, GM = Generative Models, GNN = Graph Neural Network based Models

Chapter 2

Related work

2.1 Background

This chapter addresses the fundamental papers in deep learning anomaly identification in surveillance footage that has received a lot of attention. This section was categorized based on the learning approaches found in the reviewed studies. The major theme areas include autoencoders, transfer learning, ensemble learning, reinforcement learning, and continual learning.

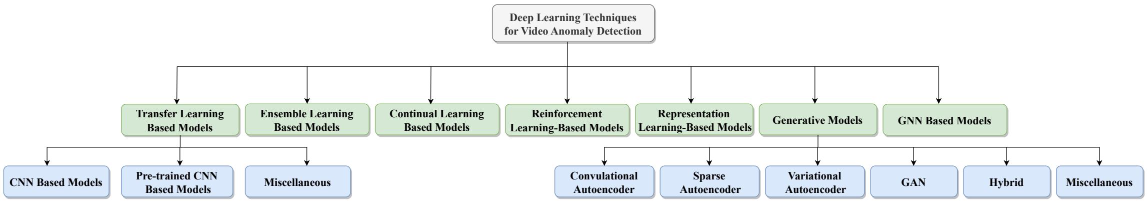

Deep Learning Techniques for Video Anomaly Detection
Transfer Learning Based Models
Ensemble Learning Based Models
Continual Learning Based Models
Reinforcement Learning-Based Models
Representation Learning-Based Models
Generative Models
GNN Based Models
CNN Based Models
Pre-trained CNN Based Models
Miscellaneous
Convulational Autoencoder
Sparse Autoencoder
Variational Autoencoder
GAN
Hybrid
Miscellaneous

Figure 2.1: The Taxonomy of Deep Learning Techniques for Video Anomaly Detection.

2.2 Literature review

2.2.1 Transfer Learning Based Models

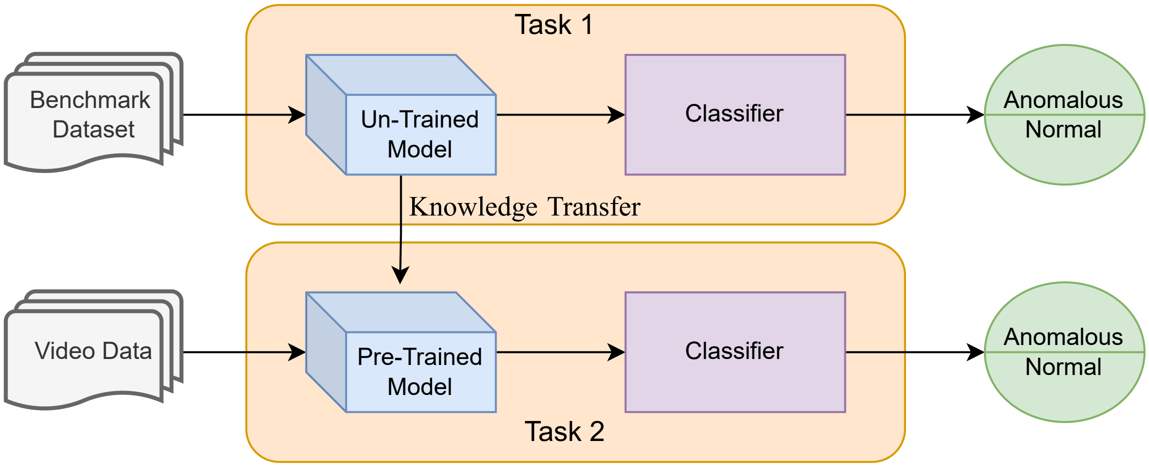


Figure 2.2: A General View of Transfer Learning-Based Models, where in Task 1, an untrained model is trained using a benchmark dataset like ImageNet, and the weights obtained from the model can be reused in Task 2, where the model will be Pre-trained and can directly involve itself in the process of predictions on a new dataset. The Process of reusing the weights by model for predictions in the unseen dataset is known as "Knowledge Transfer".

Transfer learning uses an already pre-trained model to solve a different task. When there is a scarcity of data or computing resources, transfer learning allows models to consume less data by reusing the learned weights from the pre-trained model. The benefits of transfer learning include improved performance accuracy of the base model and reduced training time. To extract characteristics from labelled video and image data, pre-trained algorithms were employed. Fig. 2.2 describes the workings of transfer learning-based models in the context of video anomaly detection. Deep 3-dimensional convolutional networks (C3D) Model, Inception V3 Module, I3D, and You Look Only Once Version 3 are the most commonly utilised pre-trained models in the studied models (YOLOV3).

CNN Based Models

Most of the preceding approaches focus on reconstructing input frames, predicting future frames, and classifying the inputted frames into anomalous or normal. Still, Sultani *et al.* [6] utilize a new deep learning approach, a weakly supervised solution that can learn anomaly patterns from both normal and anomalous videos, employing Multiple Instance Learning (MIL), which divides videos into two sections (bags), namely positive (anomalous) and negative (normal), with C3D as the feature extractor, indicates the efficacy and also underscores the importance of training with both anomalous and normal videos for a strong anomaly detection system. However, MIL still has certain issues to it such as 1) The top anomaly score in an abnormal video may not be from an abnormal snippet; 2) normal snippets randomly selected from normal videos may be relatively easy to fit, which challenges training convergence; 3) if the video contains more than one abnormal snippet, we miss the opportunity to have a more effective training process containing more abnormal snippets per video; and 4) the use of classification score provides a weak training signal that does not necessarily enable a more effective training process.5) It fails to segregate noise present in the positive section, which might cause normal snippets to be misidentified as abnormal. Several research attempts have been made, with varying degrees of success, to design effective and reasonably priced fire detection systems. Most of them, however, exhibit a trade-off between performance and model size which the work in

Pre-trained CNN Based Models

Now, when we talk about a real-world situation, it may include crowd scenes, where it is difficult to extract the required features, to solve Sabokrou *et al.* [17] uses a method where a pre-trained supervised FCN is transferred into an unsupervised FCN using fully convolutional neural networks (FCNs) and temporal data, ensuring the detection of (global) anomalies in scenes. This approach is an extension of the cascade classifier [17], where CNN is not trained from scratch but "just" fine-tuned, and in terms of input to CNN, it is a whole video instead of the cubic patch used in an earlier version. It is noticed that in both the training and testing phases, the new method is methodically simpler but faster. Furthermore, the proposed FCN is a hybrid of a pre-trained CNN (an AlexNet variant) and a new convolutional layer in which kernels are trained in relation to the training video.

Multiple Instance Learning (MIL) is used in Sultani *et al.* [6] was a major breakthrough in terms of scoring-based methods, to improve upon it, a social multiple-instance learning (MIL) framework with a dual branch network that considers dynamic interaction among groups, individuals, and environment to obtain attentive spatial-temporal feature representation is proposed in Lin *et al.* [18]. Here, the social force map is used to supply prior knowledge when modeling behavioral interaction. Furthermore, the self-attention module, which represents a more discriminative spatial-temporal feature based on the C3D network by implementing weight redistribution within the feature, is proposed. The proposed 1-D dependency capturing attention module addresses the issue that a pre-trained C3D network cannot effectively extract features. It learns various weight distributions from various positions within the feature to improve the feature's discrimination for anomaly detection.

CNN has been widely used in a different form for anomaly detection methods, and features play an important role in classifying a video, a frame, or a patch, depending on the type of input. To improve upon this, Santos *et al.* [19] suggests investigating video anomaly detection, in particular feature embeddings of pre-trained CNN that can be used with non-fully supervised data. They are also working on how to source features can be generalized for different target video domains and analyzing unsupervised transfer learning by proposing novel cross-domain generalization measures. The proposed generalization measures are not only a theoretical approach but also demonstrate practical utility as a way to understand which datasets can be used or transferred to describe video frames, allowing for better discrimination between normal and anomalous activity. For transfer learning and feature generalization experiments, six different anomaly detection videos/datasets (natural and urban scenarios) differing in several aspects, including frame resolution, amount of training frames, illumination conditions, perspectives, and presence of clutter, namely: Canoe, Boat-River, Boat-Sea, UCSD, Belleview, and Train. S. Bansod *et al.* [20] use a convolutional neural network (CNN)-based VGG16 pre-trained model to learn spatial-level appearance features for anomalous and normal patterns. Cinelli [21] proposes a real-time foreground segmentation algorithm that is competitive with the state-of-the-art for the CDNET dataset and that readily enhances and possibly surpasses the currently best methods. Nazare *et al.* [22] investigate the utility of pre-trained CNNs in anomaly detection, including VGG-16, ResNet-50, Xception, and DenseNet-121, it looks at the importance of pre-trained image classifiers in feature extraction to tackle the anomaly detection problem. And also discovers that the Xception model surpasses its competitors and may be utilized for feature extraction, despite the fact that the overall notion performs badly when compared to other anomaly detection approaches.

Ullah *et al.* [23] extracts the spatiotemporal properties from a sequence of frames by passing each one through a pre-trained convolutional neural network (CNN) model. The characteristics retrieved from the frame sequence are useful for catching unusual events. The retrieved deep features are then fed into a multi-layer Bi-directional long short-term memory (BDLSTM) model, which can effectively categorize ongoing anomalous/normal occurrences in smart city surveillance scenarios.

Khaleghi and Moin [24] also propose a new method based on deep learning techniques for anomaly detection in video surveillance cameras using CNN on the UCSD dataset. Liu *et al.* [25] explore the training samples, temporal modules for action recognition, and network backbones. More training data from surveillance videos leads to higher classification accuracy.

Accurate fall detection in the indoor environment is critical to reducing incidents of fall-related anomalies. Chhetri *et al.* [26], aims to propose a vision-based fall detection system that improves fall detection accuracy in some complex environments, such as changing light conditions in a room and the performance of video image pre-processing. The system employs the Enhanced Dynamic Optical Flow technique, which encodes the temporal data of optical flow videos using the rank pooling method, reducing processing time and improving classification accuracy in dynamic lighting conditions.

Miscellaneous

Although anomalies are generally local, occurring in a limited portion of the frame, no previous work on the subject has ever investigated the role of the locality. Therefore, in Landi *et al.* [27], the authors investigated the impact of considering spatiotemporal tubes rather than whole-frame video segments. They augmented the existing surveillance videos in the UCFCrime2Local Dataset (a subset of the UCF-Crime dataset introduced in Sultani *et al.* [6] with spatial and temporal annotations for this purpose: it is the first dataset for anomaly detection with bounding box supervision in both its train and test sets. The various experiments show that a network trained with spatiotemporal tubes outperforms a model trained with full-frame videos. Furthermore, it is observed that the locality is resistant to various types of errors during the tube extraction phase at test time. Also, the experiments demonstrated the significance of locality, our model's robustness to various types of errors, and its dependability when used to provide weak annotations on new videos.

2.2.2 Ensemble Learning Based Models

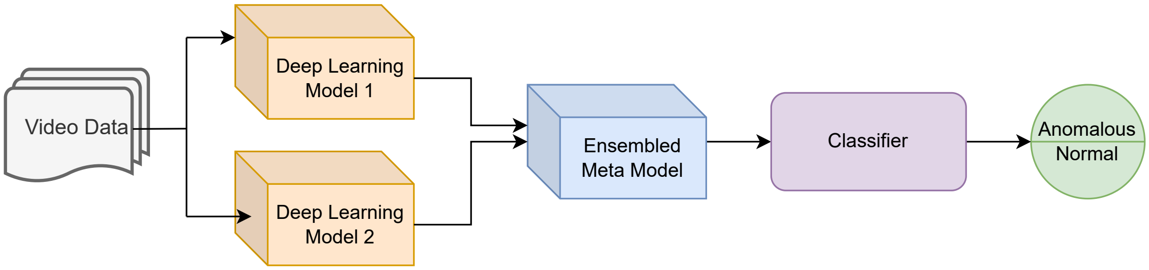


Figure 2.3: A General View of the Ensemble Learning-Based Models, where Multiple Deep Learning Models are combined to form an Ensembled Meta Model to get better predictive performance than the constituent learning model alone.

Ensemble Learning is the technique of combining numerous machine learning models to achieve superior results, Figure 2.3 shows the brief working of ensemble learning-based models, which combine the learning of different models and form one meta-ensemble model for predictions. It was observed that motion-based features are difficult to extract and also have a high false positive rate. So to resolve this, Zhu *et al.* [28] a temporal augmented network to learn a motion-aware feature. They utilized the Multiple Instance Learning (MIL) frameworks from [6] using an attention block to calculate the Area Under the Receiver Operating Characteristic Curve (AUC) while changing only the input features. The learned attention weights can aid in distinguishing between anomalous and normal video segments. They are able to achieve better results for both anomaly detection and anomalous action recognition tasks, using the proposed motion-aware feature and temporal MIL ranking model in the UCF Crime dataset compared to Sultani *et al.* [6], where it was originally proposed. Vu *et al.* [29]uses a flexible multi-channel generative framework for supervised anomaly detection in surveillance videos and various types of input images are passed into 4 CGAN streams to predict future information and apply PSNR technique to encode prediction error into feature vectors. Zahid *et al.* [30]is a typical case of ensemble learning. The model combines both a 3D convolutional network and a Fully Connected (FC) Network. Murugesan and Thilagamani [31] use a Multi-layer Perception Recurrent Neural Network (MLP-RNN) strategy based on anomaly detection classification from a video surveillance system.

2.2.3 Continual Learning-Based Models

Continual Learning refers to a never-ending learning system that gradually reinforces previously learned knowledge. When the new data to be learned differs considerably from the prior observations, the model forgets. This allows the new information to override prior knowledge in the neural network's shared internal representation. To address the catastrophic forgetting problem, a strategy to regularise the whole network in order to maintain taught information was presented.Figure 2.4 shows how

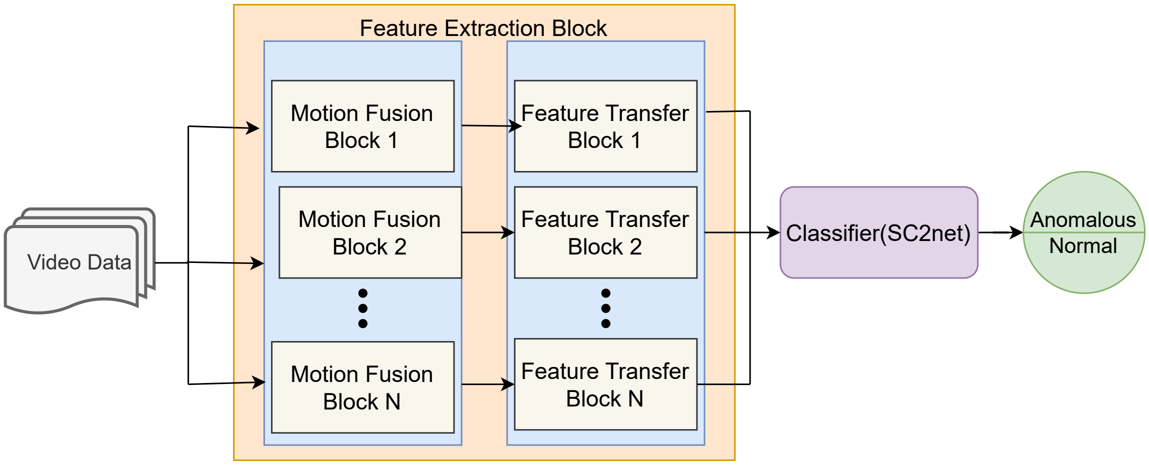


Figure 2.4: A General View of the Continual Learning-Based Model, inspired by the work in Zhou *et al.* [1], where it enforces a system of never-ending learning that gradually reinforces previously learned knowledge. Here the motion and appearance elements of an image are incorporated in the feature extraction block and its learning is passed on to the classifier for predictions.

the methodology of continual learning is being utilized in Zhou *et al.* [1].

Soon, in Zhou *et al.* [1], to combine motion and appearance features in an image, a feature learning subnetwork is used. where it is fed into a pre-trained network for feature extraction. In addition, a new subnetwork called sparse coding for the network (SC2Net) is used to compute the sparsity loss and reconstruction loss from the extracted features. Still, poor performance is possible considering the fact that the RNN and the convolutional filters encode motions and appearances separately, implying that the spatial-temporal relationships between motions and appearances are broken. Xu *et al.* [32] uses AICN an end-to-end network for anomaly detection and localization.AICN evaluates the abnormality of frames based on the intra-frame classification results. The intra-frame classification strategy reserves more connection information of sub-regions and makes the model outperform previous methods.

Doshi *et al.* [33] uses a feature extraction module and a statistical decision-making module to incrementally update the learned model within seconds using newly available nominal labels.

2.2.4 Reinforcement Learning-Based Models

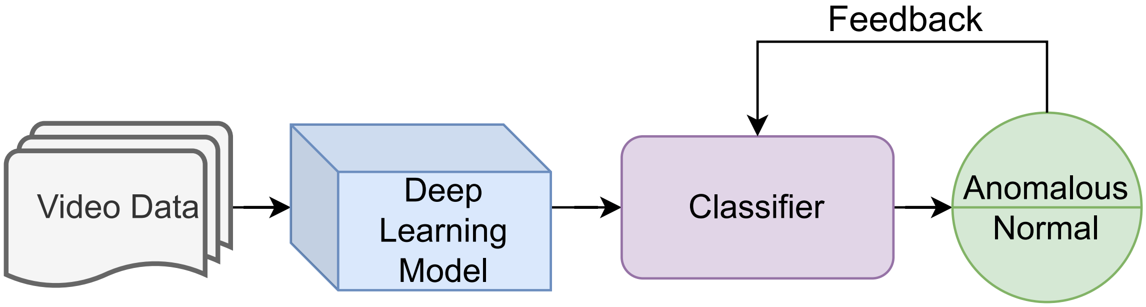


Figure 2.5: A General View of the Reinforcement Learning-Based Models, here the system operates in a continuous feedback loop where the designer can employ reward or penalty for the output. Accordingly, the classifier adjusts the weights to get desired output.

The reinforcement Learning technique describes a sequential decision-making system that uses an agent to make choices. It entails repeating a sequence of processes, the learning process is cyclic. The first stage includes an agent observing its surroundings and acquiring a new state and a reward; the second step involves the agent selecting the next course of action. The agent then sends the action to the environment, after which it adjusts its internal state based on the prior state and the actions of other agents. Figure 2.5 describes the basic workflow of various reinforcement learning-based models, which is being used for finding anomalies in videos.

It came to our attention that in previous approaches, they created patches on each video frame, and all patches were processed without checking whether or not they contained motion information. Because of the aforementioned reasons, these methods have a high computational cost and a high false alarm rate. To resolve this, Sabzailan *et al.* [34]combined the spatiotemporal convolution neural network (CNN) with handcrafted feature sets such as histograms of optical flow (HOF) and histograms of oriented gradients (HOG) for anomaly detection in contiguous video frames. Handcrafted features were learned sparsely using this novel method, Iterative Weighted Non-Negative Matrix Factorization (IW-NMF), which is based on sparse NMF. These characteristics include active volume cells, which include moving pixels in order to reduce computational costs. The CNN model's architecture allows us to extract spatial-temporal features and use handcrafted features to improve detection accuracy and robustness against local noise. Sultani *et al.* [6]propose Multiple Instance Learning that can be generalized across a variety of anomalies assigns a rating score for normal and abnormal examples using a C3D pre-trained model mixed with a light classifier, i.e. Support Vector Machine (SVM) and C3D model to extract features. The real-world UCF crime dataset is used to extract motion and trajectory elements. The model is trained on both normal and abnormal films, and the rating bags of normal and abnormal examples are generated. Every video's anomalous level is estimated using a new ranking loss function. Aberkane and Elarbi [35] identify abnormalities in videos using a Deep Q Learning Network (DQN). The model design is largely influenced by Sultani's Multiple Instance Learning [6]. The DQN enables the agent to understand how abnormalities in videos are spotted and acknowledged.DQN is made up of a completely linked layer that estimates the probability of each video clip in the anomalous and normal bags, displaying the possibility of an anomaly in a clip.

2.2.5 Representation Learning-Based Models

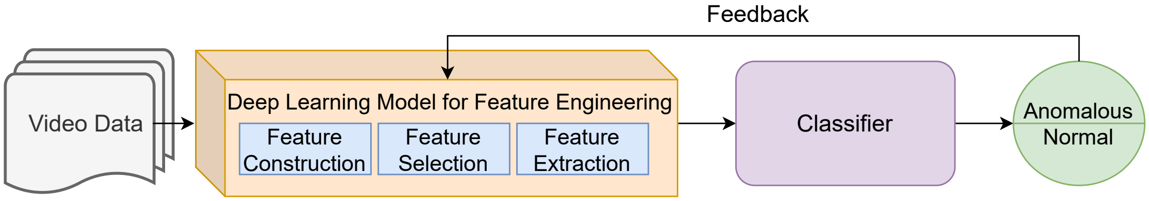


Figure 2.6: A General View of the Representation Learning-Based Models, inspired from the work in Hu *et al.* [2], where a system automatically discovers the representations required for classification from raw video data using continuous feedback. This replaces manual feature engineering by allowing a machine to learn and use features to perform a specific task.

Representation learning is a collection of techniques that allows a system to automatically identify the representations required for feature detection or classification from raw data. This eliminates the need for human feature engineering by allowing a computer to learn features and apply them to a given activity. To elucidate, the basic methodology of Representation Learning-Based Models, Figure 2.6 provides a brief overview of the general working of a model inspired by the work presented in Hu *et al.* [2]. Unlike previous works that attempted to model normal events using low-level or deep spatial-temporal features, As most of the previous method is dependent on hand-crafted features like temporal features and optical flow representation, Hu *et al.* [2] proposed a deep incremental slow feature analysis (D-IncSFA) network which is applied directly in learning progressively abstract and global high-level representations from raw data sequence, which not only detect different types of anomalies efficiently by learning feature representation from video sequences but it can also be easily applied to different datasets and extended to other sensor modalities with minimal human intervention. The main merit of the D-IncSFA network is that it has feature extractor and anomaly detector functionality, allowing anomaly detection in a single step, but it fails to identify local anomalies. Kavikul [36] proposes a CNN architecture to learn powerful features from weakly labeled data and the connectivity in architecture is able to learn spatial features.

2.2.6 Generative Models

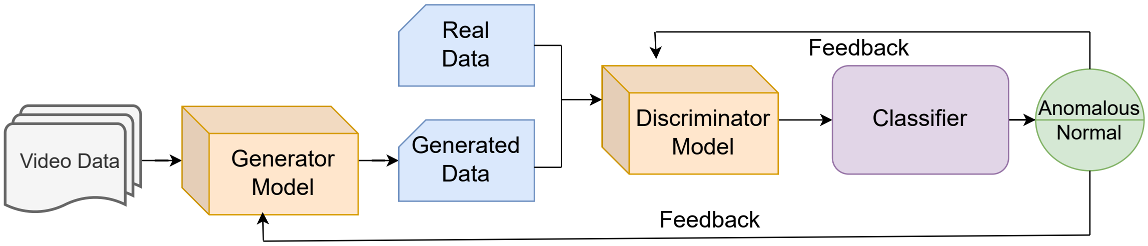


Figure 2.7: A General View of the Generative Models, where it gets trained by considering the problem as a supervised learning problem with two sub-models: the generator model, which we train to generate new examples, and the discriminator model, which attempts to classify examples as either real (from the domain) or fake (generated). Both models are part of a continuous cyclic feedback loop where they update their weights to correctly predicts the output.

In terms of a probabilistic model, a generative model specifies how a dataset is formed and can produce fresh data by sampling from this model. The generative model consists of auto-encoders, because of their unsupervised nature, capacity to train without human supervision, and unlabeled data, autoencoders are frequently employed. The key concept behind autoencoders is the reconstruction error that occurs after rebuilding the aberrant frames. The reconstruction error of irregular videos is greater than that of regular ones. This concept is used in the development of models that identify abnormalities in the video. Figure 2.7 is a very elementary take on the concept and working of Generative models, whereas the details description of various compounded models used for this task is discussed below.

Convulational Autoencoder

M.Hasan [37] uses a fully 2D convolutional autoencoder by stacking frames in channels to model regular frames, which captures temporal appearance changes since it takes a short video clip as input. He learned the model of normal behaviors using deep learning-based autoencoders. He detected abnormalities using reconstruction loss, capturing the regularities from multiple datasets like CUHK Avenue, Subway, and UCSD Pedestrian datasets, preserving Spatiotemporal information. Still, it only sometimes retains temporal information. It considered only normal behaviors for detection using weakly labeled training data. Also, it learns common patterns very quickly. It still yields high anomaly ratings even for new standard patterns. In Reality, CNNs were not designed with temporal features in mind and are therefore unsuitable for video. Therefore, Medel [38] implemented the approach, where they trained generative models to detect anomalies in videos with minimal supervision. They proposed an End-to-end trainable composite Convolutional Long Short-Term Memory (Conv-LSTM) network that can predict the evolution of a video sequence from a small number of the input frame. Here, the reconstruction errors of a set of predictions with abnormal video sequences yielding lower regularity scores as they diverge further from the actual sequence over time are used to calculate regularity scores. The models also employ a hybrid structure and investigate the effects of 'conditioning' in learning more meaningful representations. This composite model suffers from a few challenges like they do not adapt to new (unusual) movements and overtime errors in object prediction were observed, which replaced moving objects with pedestrians.

From the anomaly detection perspective, the Convolutional Autoencoder (CAE) is still an interesting choice, since it captures the 2D structure in image sequences during the learning process and it was clearly described in Hasan *et al.* [37], whose works have been extended in Ribeiro *et al.* [39] by including low-level features such as optical flow and edges as inputs alongside the raw frames for the Convolutional Autoencoder (CAE). The following work employs a CAE in the context of anomaly detection, using the reconstruction error of each frame as an anomaly score. The proposed method aggregates high-level spatial and temporal features with the input frames and investigates how they affect CAE performance while exploring the CAE architecture, also a simple measure of video spatial complexity was developed and correlated with the CAE's classification performance.

The majority of the previously mentioned methods extract either global or local features without taking the objects of interest into account. To facilitate this Ionescu *et al.* [40], developed an unsupervised feature learning framework based on object-centric convolutional auto-encoders to encode both motion and appearance information. They also proposed a supervised classification method by grouping training samples into normality clusters. Furthermore, it can be improved by segmenting and tracking objects. The Spatial-temporal autoencoder by Chong and Tay [41] is different due to its building constructs. It employs time-distributed layers wrapped in conv2d layers for the spatial part and convlstm2d for the temporal part

Sparse Autoencoder

To get improved performance and improved run-time performance over previous methods, Sabokrou *et al.* [17] in 2016 proposed a two-stage cubic-patch-based cascade classifier to perform anomaly detection and localization of anomalous regions, with these two stages built on analyzing the Reconstruction error (RE) and Sparsity Value (SV). The proposed method is a quick, simple, and accurate method for locating abnormal events in the video by leveraging the RE of AE and its ability to provide a sparse representation of a normal patch. The dependency of preceding methods on low-level features not only affected the performance and computation time of the model but also decreased the applicability of the model, also they suffer from a high false-positive rate and are not real-time, rendering them practically obsolete.

The autoencoder developed by Duman and Erdem [42] is made up of Convolutional Autoencoder and Convolutional LSTM. This system extracts speed and trajectory characteristics from movies using Optical Flow. The autoencoder receives the optical flow output and returns the reconstructed optical flow map. The reconstructed output is subtracted from the input to provide the mean squared error, which is used to produce the regularity score, which represents the extent of irregularity in each frame. Ramchandran and Sangaiah's [43] propose an unsupervised solution for anomaly detection in crowded scenes. Conv-LSTM is used to build the model. The model is trained using raw picture sequences and edge image sequences. Bhakat and Ramakrishnan [44] use a spatial-temporal autoencoder, it is distinct owing to its construction. It uses time-distributed layers wrapped in conv2d layers for the spatial component and convlstm2d for the temporal component.

Variational Autoencoder

Pawar *et al.* [45], proposed an unsupervised approach learning approach based on deep learning and one class learning paradigm for the detection of global anomalies from crowd surveillance videos.

GAN

Nguyen *et al.* [46], proposes a novel method based on the GAN approach to detect abnormal events in traffic surveillance videos. The method predicts the following frames from a video using U-net architecture. To handle fast-moving objects in traffic videos, it proposes to replace the raw input of a sequence of frames into the generator network with the blended motion descriptor using an average image. The average image can capture information on moving up to the current frame, and provide stacked motion information for the next frame generation in normal scenarios. It also proposes a refined loss function to focus on the quality of a vehicle's boundary, by comparing the difference between a generated frame and a real next frame, we can identify an anomaly

Hybrid

To address these shortcomings, dynamic anomaly detection and localization system are proposed by Narasimhan *et al.* [47], which employs deep learning to learn relevant features automatically. Each video is represented as a group of cubic patches in this technique for detecting local and global anomalies. The main advantage of this method is the reduction in execution time, which significantly shortens the time required to detect anomalies in large videos with many frames. Another approach that improves the computational time required for anomaly detection is by Sabokrou *et al.* [48], who extend their work on cubic-patch-based cascade classifiers, where deep 3D autoencoder is used in the first stage and deeper 3D CNN is used at the second stage, where the Shallow layers of cascaded deep networks (designed as Gaussian classifiers acting as weak single-class classifiers) detect ”simple” normal patches such as background patches, while deeper layers detect more complex normal patches. While evaluating the results it was observed that on standard benchmarks, it performs similarly to top-performing detection and localization methods around that time but outperforms them in terms of required computation time.

Real-world anomalies are not restricted to appearance anomalies or unnatural motion anomalies, so to create a highly efficient anomaly detection system, a dataset compromise of ground truth situations was needed, which was introduced in Zhao *et al.* [49], along with their novel model called Spatio-Temporal AutoEncoder, which uses deep neural networks to learn video representation automatically and extracts features from both spatial and temporal dimensions using 3D convolutions in the encoder and 3D deconvolution in the decoder. It also introduced a weight-decreasing prediction loss for predicting future frames, which guides the model to capture the trajectory of moving objects and forces the encoder to extract the temporal features better.

To address this issue, a new autoencoder was introduced in Gong *et al.* [50]called memory-augmented autoencoder (MemAE) which can improve the performance of the autoencoder-based unsupervised anomaly detection methods and that can store encodings in memory. Given an input, MemAE proposes the use of an encoder to obtain an encoded representation and then use the encoding as a query to retrieve the most relevant patterns in memory for reconstruction. Because the memory has been trained to record prototypical normal patterns, the proposed MemAE can well reconstruct the normal samples while increasing the reconstruction error of the anomalies, thereby strengthening the reconstruction error as an anomaly detection criterion.

Fan *et al.* [51], proposed an efficient partially supervised deep-learning methodology for detecting and locating anomalous events in surveillance videos. Our method is based on a two-stream network framework that uses RGB frames and dynamic flows, respectively. Image patches of normal samples from each stream are extracted as input to train a Gaussian Mixture Fully Convolutional Variational Autoencoder (GMFC-VAE) that learns a Gaussian Mixture Model (GMM). The conditional probabilities of each component of a Gaussian Mixture of test patches are obtained in the testing stage by employing the GMFC-VAE for each stream. We present a sample energy-based method for predicting a score for appearance and motion anomalies. However, the work is only validated in public video sequences captured by a fixed camera. Because of the scarcity of massively similar normal events in more complex conditions, such as changing scenes, learning the proper estimation of distribution parameters is difficult. One possible solution is to constantly optimize the detection model based on newly observed normal events in order to avoid variance collapse in some space regions with poor representation of normal samples. Another approach is to consistently train the detection model using Reinforcement Learning (RL) and reward the detection result.

Miscellaneous

Almost all reconstruction-based methods address the issue by minimizing training data's reconstruction errors, which cannot guarantee a larger reconstruction error for an abnormal event. Lately, In Liu *et al.* [52], addresses the anomaly detection problem within the context of a video prediction framework. This is the first work to detect an abnormal event by leveraging the difference between a predicted future frame and its ground truth. Along with that, it enforces the optical flow between predicted frames to be close to their optical flow ground truth in the video frame prediction framework, in addition to forcing predicted frames to be close to their spatial ground truth. They also introduced a next dataset called the Toy dataset and experiments on it validate the robustness to uncertainty for normal events, which validates the robustness of our method. Here U-Net is used as our basic prediction network. Most of the previous work on reconstruction assumes that the anomalous instances will have a large reconstruction error. But, this assumption does not necessarily hold true always, primarily because an autoencoder may be able to generalize well in some cases. This is problematic because it may accurately reconstruct anomalous instances as well.

2.2.7 GNN Based Models

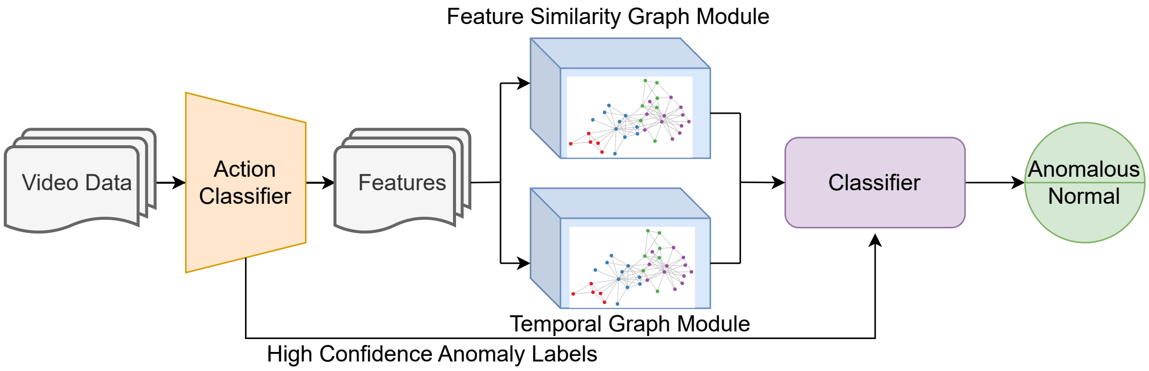


Figure 2.8: A General View of the Graph Neural Network-based Models, inspired by the work in Zhong *et al.* [3], where firstly the action classifier extracts spatiotemporal features from erroneous video frames and generates noisy frame-level labels, then the frame-level features are compressed and fed into feature similarity and temporal consistency graph modules. These two models' outputs are combined and used to predict frame-level predictions. The loss is updated using high-confidence frame labels to correct the predictive noise.

Graph Neural Networks (GNNs) are a type of deep learning approach that performs inference on graph-described data. GNNs are neural networks that may be applied directly to graphs to perform node-level, edge-level, and graph-level prediction tasks. Figure 2.8 describes the overview of the methodology, which is partially inspired by the work in Zhong *et al.* [3], where GNN acts as a label noise cleaner in the process to predict whether the frame is anomalous or normal.

In Zhong *et al.* [3], addresses weakly supervised anomaly detection from a new perspective, by casting it as a supervised learning task under noise labels. In contrast to MIL formulation in previous works, such a perspective possesses distinct merits in two aspects: a) it directly inherits all the strengths of well-developed action classifiers; b) anomaly detection is accomplished by an integral end-to-end model with great convenience. Furthermore, we utilize a GCN to clean labels for training an action classifier. During the alternate optimization process, the GCN reduces noise via propagating anomaly information from high-confidence predictions to low-confidence ones. It validated the proposed detection model on 3 different-scale datasets with 2 types of action classification networks, where the superior performance proves its effectiveness and versatility

In general, the current research on video anomaly detection has several limitations. Firstly, some approaches either ignore valuable optical flow information or use datasets with a limited number of anomalies and data. Secondly, the employed loss functions are not suitable for the perfect distinction between normal and anomalous events. Thirdly, many existing methodologies include models with high time complexity, which is unsuitable for real-time video anomaly detection. To overcome these limitations, new methods should be developed that incorporate optical flow information, use large datasets for training, and develop novel loss functions or modify existing ones. Additionally, time-efficient models should be used. Addressing these limitations has the potential to significantly improve video anomaly detection.

Table 2.1: A summary of various deep learning techniques used for Video Anomaly Detection

Author	**Learning Technique**	**Deep Learning Algorithm/Models**	**Datasets**	**Performance**
Doshi & Yilmaz [33]	Continual Learning	YOLOv3 KNN	UCSD, Avenue, Shangai Tech	85%(ACC)
Zhou *et al.* [1]	Continual Learning	AnomalyNet: a unified approach	CUHK avenue, UCSD Pedestrian and UMN	86.1%(AUC)
Zhong et al. [3]	GNN Based Models	TSN RGB	UCF-Crime,ShanghaiTech and UCSD-Peds	82.12%(ACC)
Sultani *et al.* [6]	Reinforcement Learning	MIL method for anomaly detection	UCF-Crime	75.41%(ACC)
Sabzailan *et al.* [34]	Reinforcement Learning	Spatial-temporal CNN	UCSD and UMN	99.6%(AUC)
Aberkane and Elarbi [35]	Reinforcement Learning	Deep Q Learning Network (DQN)	UCF Crime	n/a
Zhu *et al.* [28]	Ensemble Learning	Temporal augmented network with motion-aware feature	UCF Crime dataset	79% (AUC)

Table2.1 – *Continued from previous page*

Author	**Learning Technique**	**Deep Learning Algorithm/Models**	**Datasets**	**Performance**
Vu *et al.* [29]	Ensemble Learning	R-CNN, SVM,CGAN	Avenue,UCSD Ped1, Ped2,Shanghai Tech	91.7% (ACC)
Zahid and others [30]	Ensemble Learning	Fully Connected Network, Inception V3	UCF Crime101	92.06%(ACC)
Murugesan and Thilagamani [31]	Ensemble Learning	MLP-RNN	Custom Dataset	98.56%(ACC)
Hu *et al.* [2]	Representation Learning	Deep Neural Network + Slow Feature Analysis	UMN and PETS 2009	96.92%(AUC)
Kavikuil & Amudha [36]	Representation Learning	CNN	Avenue	0.99 (F1 Score)
Landi *et al.* [27]	Transfer Learning	3D-CNN	UCF-Crime	74.73%(AUC)
Sabokrou *et al.* [48]	Transfer Learning	Deep-Anomaly	UCSD and Subway	90.2%(AUC)

Table2.1 – *Continued from previous page*

Author	**Learning Technique**	**Deep Learning Algorithm/Models**	**Datasets**	**Performance**
Lin *et al.* [18]	Transfer Learning	Multiple-Instance Learning + Social Force Maps	UCFCrime dataset	78.28%(AUC)
Santos *et al.* [19]	Transfer Learning	Transfer Component Analysis	UCSD Pedestrian 1 (Ped1) ,UCSD Pedestrian 2 (Ped2) and Custom	63.84%(AUC)
Nasaruddin [53]	Transfer Learning	3D-CNN	UCF Crime	99.25% (ACC)
Doshi *et al.* [54]	Transfer Learning	Pre-trained Convnet (YOLOV3) & Least Square Generative Adversarial Network LS-GAN	UCSD Ped2 & CUHK Avenue	84.83% (ACC)
Ullah *et al.* [23]	Transfer Learning	Pre-trained CNN, BD-LSTM	UCF Crime	89.05%(ACC)
M.Hasan *et al.* [37]	Generative Models	Fully 2D Convolutional Autoencoder	Avenue, UCSD pedestrian, and Subway	83.18%(ACC)

Table2.1 – *Continued from previous page*

Author	**Learning Technique**	**Deep Learning Algorithm/Models**	**Datasets**	**Performance**
Medel [38]	Generative Models	Convolutional Long Short-Term Memory(Conv-LSTM)	UCSD Pedestrian, Avenue Dataset and Subway	0.659 / 0.967(Precision /Recall)
Sabokrou *et al.* [17]	Generative Models	Sparse Autoencoder +Autoencoder	UCSD and Subway datasets	99.6%(AUC)
Narasimhan *et al.*[47]	Generative Models	Sparse Denoising Autoencoders	UMN dataset and UCSD Pedestrian dataset	99.6%(AUC) 2.2 (EER)
Sabokrou *et al.* [55]	Generative Models	Cascade of Deep Convolutional Neural Networks + Autoencoders	UCSD and UMN	99.6%(AUC)
Zhao *et al.* [49]	Generative Models	Spatiotemporal Autoencoder	UCSD Pedestrian, CUHK Avenue and Traffic (Custom made)	86.8%(AUC)

Table2.1 – *Continued from previous page*

Author	**Learning Technique**	**Deep Learning Algorithm/Models**	**Datasets**	**Performance**
Ribeiro *et al.* [39]	Generative Models	Low-level Features + 2D Convolutional Autoencoder	UCSD pedestrian dataset (including the two subsets, Ped1 and Ped2) and Avenue Dataset	73.2%(AUC)
Liu *et al.* [52]	Generative Models	Future Frame using U-Net	CUHK, UCSD Dataset and ShanghaiTech, Toy pedestrian dataset(custom)	84.05%(AUC)
Gong *et al.* [50]	Generative Models	Autoencoder + memory module + attention-based	UCSD-Ped2 , CUHK Avenue and ShanghaiTech addressing	94.10%(AUC)
Ionescu *et al.* [40]	Generative Models	Object-Centric Convolutional Autoencoders	Avenue, ShanghaiTech, UCSD and UMN	99.6%(AUC)

Table2.1 – *Continued from previous page*

Author	**Learning Technique**	**Deep Learning Algorithm/Models**	**Datasets**	**Performance**
Fan *et al.* [51]	Generative Models	Gaussian Mixture Fully Convolutional Variational Autoencoders	UCSD Dataset and The Avenue Dataset	84.17%(AUC)
Duman *et al.* [42]	Generative Models	OF-ConvAE-ConvLSTM	Avenue UCSD Ped1, Ped2	91.53%(ACC)
Nguyen [46]	Generative Models	Conv-Net, GAN	UCSD Ped2, CHUK Avenue, Subway Entrance, Exit	91% (ACC)
Ramchandran *et al.* [13]	Generative Models	ConvLSTM	UCSD Ped1 & Ped2	84.7% (AUC)
Pawar *et al.* [45]	Generative Models	ConvAE, LSTM AE	UMN	87%(AUC)
Chong & Tay [41]	Generative Models	Auto-encoder ConvLSTMAE	UCSD Ped1, UCSD Ped2	87% (ACC)
Bhakat and Ramakrishnan [44]	Generative Models	ConvLSTM	Avenue, Surveillance Office, Police	73.6%(ACC)
Doshi *et al.*[56]	Transfer Learning	Pre-trained Convnet (YOLOV3) & GAN	UCSD PED2, CUHK, ShanghaiTech	84.87%(ACC)

Table2.1 – *Continued from previous page*

Author	**Learning Technique**	**Deep Learning Algorithm/Models**	**Datasets**	**Performance**
Cinelli [21]	Transfer Learning	Pre-trained CNN ResNet	CDNET2014	85%(ACC)
Nazare, Mello & Ponti [22]	Transfer Learning	Pre-trained CNNs	UCSD Ped2	76%(ACC)
Bansod & Nandedkar [20]	Transfer Learning	Pre-trained CNN (VGG16)	UCSD, UMN	91.4%(ACC)
Liu *et al.* [25]	Transfer Learning	Binary Networks, 3DCNN	citySCENE	94.6%(ACC)
Khaleghi and Moin [24]	Transfer Learning	CNN	UCSD	EER Frame 14% EER Pixel 25%
Xu *et al.* [32]	Continual Learning	Adaptive Intra-Frame Classification Network	UCSD Ped1 dataset, UCSD Ped2 dataset, Avenue dataset and Subway dataset	90.8%(AUC)

2.3 Benchmark Datasets

2.3.1 ShanghaiTech Dataset

The ShanghaiTech [52] dataset was generated at ShanghaiTech University in 13 scenarios with complicated lighting and camera views. It is made up of 437 videos, each containing 726 average frames. The training set includes 330 normal films, whereas the testing set includes 107 videos with 130 abnormalities. Anomaly occurrences involve unexpected campus patterns, such as bicyclists or cars.

2.3.2 UCF Crime Dataset

UCF Crime [6] is a dataset of 1900 uncut films that highlight 13 real-world aberrant occurrences, including Abuse, Arrest, Arson, Assault, Road Accident, Burglary, Explosion, Fighting, Robbery, Shooting, Stealing, Shoplifting, and Vandalism. There are 950 regular videos and the remaining videos each have at least one abnormality occurrence. The training set includes 800 normal films and 810 aberrant videos. For validation, the remaining 150 normal and 140 aberrant films are chronologically tagged. Both the training and testing sets include all 13 anomalous events. Some videos may contain numerous oddity categories, such as robbery and fighting, burglary and vandalism, and arrest and gunfire. All of the videos are suitable for real-world surveillance. Furthermore, UCF Crime includes a wide range of lighting conditions, image resolutions, and camera angles in complicated circumstances, making it extremely difficult.

2.3.3 Avenue Dataset

The Avenue [57] dataset has 15 videos, each of which is around 2 minutes long. The total number of frames is 35,240. As a training set, 8,478 frames from four videos are used,the dataset contains 47 different anomalies which include loitering, running, and throwing objects

2.3.4 UMN Dataset

The UMN dataset (University of Minnesota) consists of five videos recorded from various viewpoints. These video clips were captured at 30 frames per second using a stationary camera that has no significant illumination changes. With respect to the number of frames, all in all there are 7,740 frames where 1,450, 4,415, and 2,145 belong to lawn, indoor, and plaza scenes, respectively.In this dataset, the particular anomaly that happens is when the people run to escape or when they panic. The sequences generally start with normal behavior where an escape panic behavior ensues.

2.3.5 UCSD Dataset

The UCSD dataset is divided into two sections, Ped1 and Ped2. They are captured at two locations on the UCSD campus where most people walk. The training set comprises only normal frames (34 clips for Ped1 and 16 clips for Ped2), whereas the test set contains both normal and anomalous frames (36 clips for Ped1 and 12 clips for Ped2). All test clips include frame-level annotation, and ten of them have pixel-level ground truth. There are around 3,400 frames with anomalies present while the normal frames are around 5,500. Both subsets have a frame-level ground truth and a pixel-level ground truth.

2.3.6 Subway Dataset

The subway dataset is divided into two subsets: Subway Entrance and Subway Exit. Each subway stop has only one lengthy surveillance video. They are originally recommended for real-time identification of unexpected events in busy subway settings, such as traveling in the opposite direction or without paying. The exit gate video has 136,524 frames while the entrance gate video has 72,401 frames.

Table 2.2: Comparison of different Datasets

Datasets	**Dataset Description**	**Dimensionality**	**Best Performing Model**	**Performance***
ShanghaiTech[1] [52]	Chasing, Brawling Sudden Motion, etc	856×480	S3R [58]	97.48% (AUC)
UCF Crime [2] [6]	Assault, Burglary, Robbery, etc	320×240	S3R [58]	85.99 % (AUC)
Avenue [3] [57]	Loitering, Running, Throwing objects	640×360	SSMCTB [59]	93.2% (AUC)
UMN [4]	Escape Panic	320×240	Object Centric Convolutional Autoencoder [40]	99.6% (AUC)
UCSD Ped 1 [5]	Carts, Bikers, Walking	238×158	DeepGAN	98.5% (AUC)
UCSD Ped 2 [6]	Carts, Bikers, Walking etc	360×240	RTFM [60]	98.6% (AUC)
Subway [7]	Loitering, Wrong direction walking, Irregular interactions among people, No payments etc	512×384	ConvLSTM-AC [61]	94.05% (AUC)

[1] https://svip-lab.github.io/dataset/campus_dataset.html

[2] https://www.crcv.ucf.edu/projects/real-world

[3] http://www.cse.cuhk.edu.hk/leojia/projects/detectabnormal/dataset.html

[4] http://mha.cs.umn.edu/

[5] & [6] http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm

[7] https://www.kaggle.com/datasets/new-york-state/nys-nyc-transit-subway-entrance-and-exit-data

Chapter 3

Proposed Method

3.1 Overview

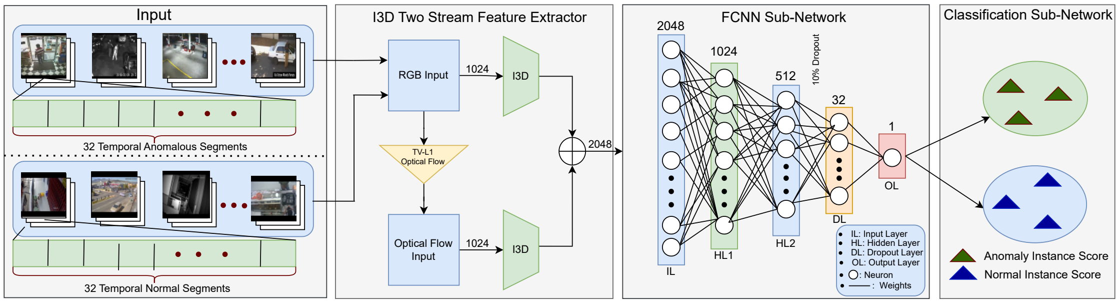


Figure 3.1: A complete pipeline of the proposed solution in a step-wise manner, step 1 represents input sub-network, where inputs from normal and anomalous videos are divided into 32 segments and are provided as an input to step 2, which is a two-stream I3D feature extractor, where both RGB frames and optical flow information of the video is utilized to extract features and provided to the 5 layered fully connected neural network, which uses modified MIL ranking loss function in step 3 perform classification

The proposed approach is divided into 3 phases as described in Figure 3.1 and the discussion regarding each phase is as follows:

3.1.1 Phase 1

Input

UCF-Crime [6]is a large-scale anomaly detection dataset that comprises 1900 untrimmed films from real-world street and indoor surveillance cameras with a total duration of 128 hours. There are the same amount of normal and anomalous videos in both training and testing sets. With 1,610 training videos with video-level labels and 290 test videos with frame-level labels, the data set contains 13 groups of anomalies such as abuse, arrest, arson, assault, road accident, burglary, explosion, fighting, robbery, shooting, stealing, shoplifting and vandalism.

Most of the datasets used for Video Anomaly detection had some form of limitations in terms of the Number of Videos they contain and they don't have diversity in terms of the type of anomalies to detect. To overcome the above limitations, Sultani et. al [6]proposed a new large-scale dataset, which includes long untrimmed surveillance videos that cover 13 different anomalies, such as Arson, Assault, Burglary, and Vandalism. These anomalies were chosen because they pose a significant threat to public safety. Comparison of the dataset with previous ones in Table C.1.

To ensure the high quality of the dataset, ten annotators with varying levels of computer vision expertise were employed to collect data using text search queries such as "car crash" and "road accident" on YouTube and LiveLeak. Also, the text queries in multiple languages were translated using Google Translate, to maximize the number of videos collected. Different pruning processes such as removing manually edited, prank, non-CCTV captured, news-sourced, handheld camera-captured, and compilation videos, as well as those where the anomaly was unclear, were utilized to get better inputs. This resulted in a collection of 950 real-world surveillance videos featuring clear anomalies and another 950 normal videos, totaling 1900 videos in the dataset.

In terms of annotations, the original dataset contains only temporal annotations i.e. Frame level annotations, which were gathered with the help of multiple annotators, also only the testing split of the dataset contains temporal annotations, no annotations were provided for the training split of the dataset.

Our dataset is divided into two parts: the training set, which includes 800 normal and 810 anomalous videos, and the testing set, which includes the remaining 150 normal and 140 anomalous videos. Both the training and testing sets contain all 13 anomalies in the videos at various temporal locations. In addition, some of the videos contain multiple anomalies.

3.1.2 Phase 2

Pre-Processing

1. **Resizing**: The first step is to resize the input video frames to a fixed size to ensure uniformity across all frames.

2. **Center and Random Cropping**: The second step is to perform cropping on the video frame, where in center cropping the center of the video frame is

cropped out, and the remaining part is used for processing and random cropping involves randomly selecting a section of the image or video and cropping it out to create a new image or video sample

3. **Normalization**: The next step is to normalize the pixel values of the cropped video frames. This is done to ensure that the data has zero mean and unit variance, which helps in better convergence during training.

4. **RGB to BGR Conversion**: In the two-stream I3D model, the input frames are expected to be in the BGR format, which is different from the RGB format used by most video codecs. Therefore, the RGB frames are converted to BGR format before feeding into the model.

5. **Optical Flow Computation**: For the second stream of the two-stream I3D model, optical flow computation is performed on the input video frames. Optical flow is the pattern of apparent motion of objects in a video, which is computed by comparing adjacent frames. It helps in capturing the temporal information of the video.

6. **Optical Flow Normalization**: Finally, the optical flow values are normalized to have zero mean and unit variance, similar to the RGB frames, to ensure consistency between the two streams.

Feature Extraction

Raw videos can't be used for anomaly prediction, so there is a need to extract the important features from them, and then give them to the Fully Connected Neural Network for Anomaly score prediction. For the feature extraction purpose, There has

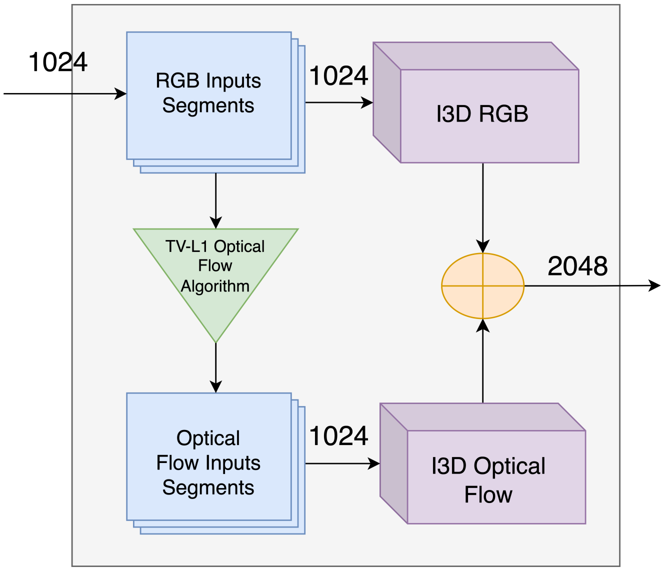


Figure 3.2: A brief overview of two stream feature extraction process, where the RGB frames of 1024 dimension plus the optical flow information of also 1024 dimension, which is computed using the TV-L1 optical flow algorithm[4], where RGB frames are provided as input. At last, both of them are concatenated and provided as input of dimension 2048 to the FCNN.

been the utilization of Two-Stream Inflated 3D ConvNet (I3D) based on 2D ConvNet inflation, where deep image classification filters and kernels ConvNets are extended to 3D, allowing for the learning of seamless spatiotemporal features from video while leveraging successful ImageNet architecture designs and even their parameters. It was observed that, after pre-training on Kinetics, I3D models outperform the state-of-the-art methods in action classification.

The I3D model employs a two-stream architecture, with one stream handling RGB frames and the other handling optical flow frames as shown in Figure 3.2. Before merging, each stream has separate convolutional and pooling layers and it is made up of 25 layers, including 9 Inception modules[5]. Inception modules are used for efficient feature extraction by concatenating multiple filter sizes in parallel. The detailed structure of the I3D model is illustrated in Figure 3.3.

Data augmentation is also critical for improving I3D's performance. As discussed in Pre-Processing section, used random cropping during training, which involved resizing the smaller video side to 256 pixels and randomly selecting a 224×224 patch, as well as temporal cropping, which involved selecting the starting frame among those that were early enough to ensure the desired number of frames. For shorter videos, repeated the video until it met the model's input requirements. Furthermore, during training, then applied random left-right flipping to each video. The models were applied convolutionally to the entire video during testing, taking 224×224 center crops and averaging the predictions.

The novel part of our approach is that the two-stream I3D model has never been used for Feature extraction for Video Anomaly detection, In the project original model was taken and modified for the UCF-Crime dataset to extract better features for anomaly detection. The original implementation is done in TensorFlow [1], while there has been modification and implementation in PyTorch [2]

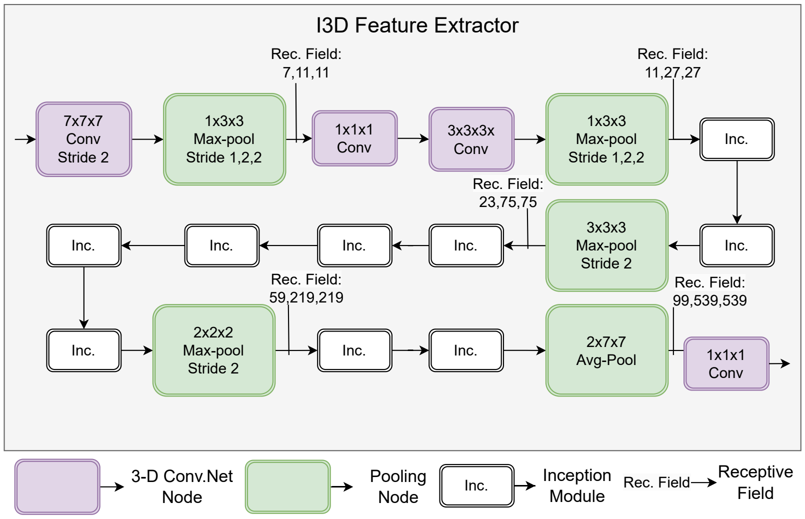


Figure 3.3: A Detailed overview of the I3D feature extractor model architecture[5].

3.1.3 Phase 3

Model Architecture

There occurs splitting of each video into 32 distinct segments to improve performance and treat each segment as a bag instance just like Sultani *et. al.* [6]. Despite experimenting with overlapping temporal segments of different scales, But found that they did not enhance detection accuracy. As a minibatch, Then there is random selection of 30 positive and 30 negative bags.

As, clearly described in Figure 3.4, Then the input of the extracted features of 2048-Dimension to 5 Layer Fully connected neural network. The first 3 layers are followed by the ReLU Activation function [?], because it introduces non-linearity, produces sparse activations, is computationally efficient, and helps to avoid the van-

[1] https://www.tensorflow.org
[2] https://pytorch.org

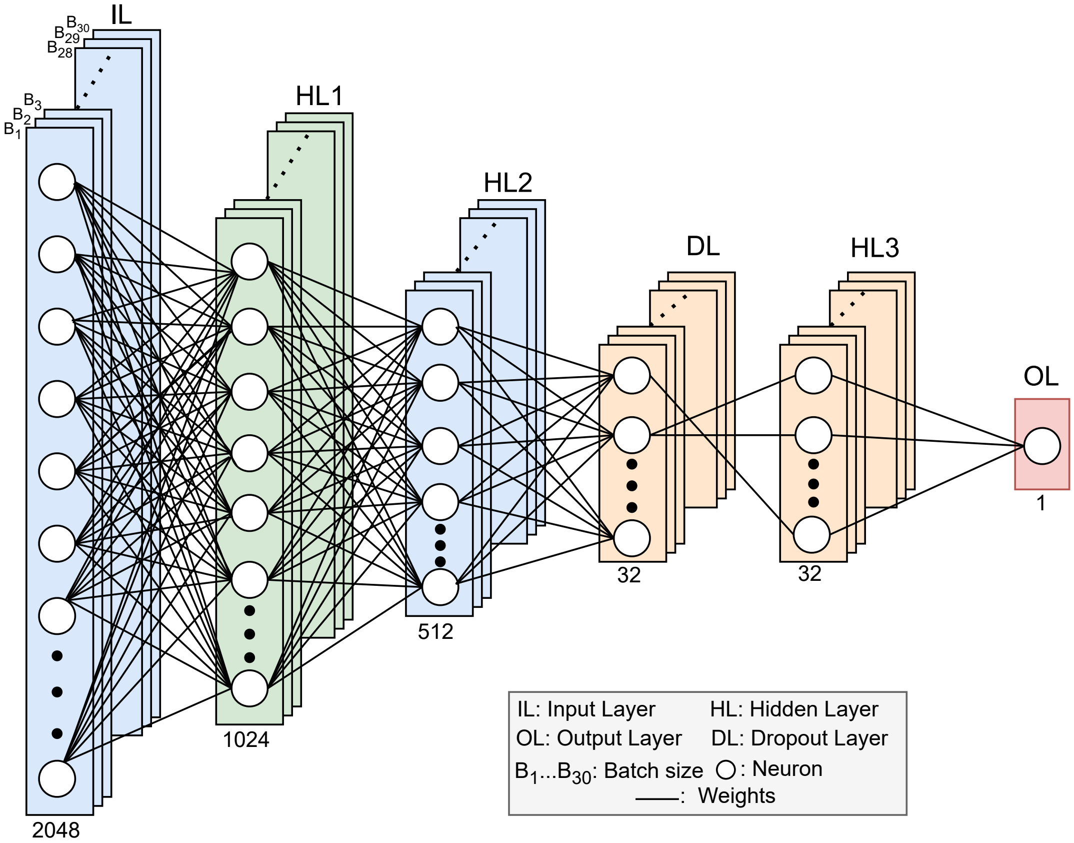


Figure 3.4: Detailed layer-wise schematic of proposed 5-layered fully connected neural network model, which consists of one input layer represented by IL, three hidden layers represented by HL, of which HL3 is also a Dropout layer represented by DL, and an output layer represented by OL. All layers are followed by ReLU Activation, whereas OL is followed by Sigmoid Activation.

ishing gradient problem. It is applied to the output of each neuron in the hidden layers and is defined in Equation (3.1). The last layer is followed by Sigmoid Activation Function [?] defined in the Equation (3.2), as the output of the sigmoid function ranges from 0 to 1, with values close to 0 indicating normal inputs and values close to 1 indicating abnormal inputs, also, the sigmoid function a smooth gradient, which makes it a good choice for backpropagation algorithms that are used to train neural networks. Then usage of 10% Dropout Regularization in the 4^{th} layer to prevent overfitting in neural networks. It can be applied by randomly dropping out (setting to zero) 10% percentage of the neurons in a layer during each training iteration; it is relatively a high dropout rate as the model contains a large number of parameters.

$$f(x) = \max(0, x), \tag{3.1}$$

where x is the input to the activation function and f(x) is the output.

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \tag{3.2}$$

where x is the input to the function and it can take any real value, and the function will output a value between 0 and 1.

The Xavier initialization method is used, as described in Equation (3.3) for initializing the weights of the linear layers in our network, which is a crucial technique in deep learning used for weight initialization in neural networks. It helps to keep the variance of activations and gradients consistent across layers, preventing issues like vanishing and exploding gradients which can lead to unstable training and hinder the network's ability to learn effectively. It can result in faster convergence, improved accuracy, and a more stable training process. Hence, it is widely used in the initialization of deep neural network weights to enhance their performance.

The Xavier initialization formula for a layer with n inputs and m outputs is:

$$W \sim U\left(-\frac{\sqrt{6}}{\sqrt{n+m}}, \frac{\sqrt{6}}{\sqrt{n+m}}\right), \tag{3.3}$$

where W is the weight matrix and U is a uniform distribution. The formula ensures that the weights are initialized to values that have a variance of $1/n$ and $1/m$ for the input and output layers, respectively. This helps to prevent the gradients from vanishing or exploding during training.

Model Training and Hyperparameters

Table 3.1: Hyperparameter Table

Hyperparameter	Value
Optimizer	Adam
Batch size	30
Loss function	MIL
Learning rate	0.001
Weight decay	1×10^{-12}
Number of epochs	100

Table 3.1 shows various hyperparameters used in our approach and their significance is as follows:-

1. **Optimizer**: Adam Optimizerr [3], is utilized as described in Equation (3.4) for our final model training, as it helps the model, learn to make better predictions, adjust the weights of the model during training, and speed up convergence on our large dataset.

   The Adam optimization algorithm updates the parameters of a neural network based on the first and second moments of the gradients. The formula for Adam is:

[3] https://pytorch.org/docs/stable/generated/torch.optim.Adam.html

$$
\begin{aligned}
m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\
\hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\
\hat{v}t &= \frac{v_t}{1-\beta_2^t} \\
\theta_t &= \theta t - 1 - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t,
\end{aligned} \tag{3.4}
$$

where m_t and v_t are the first and second moments of the gradients at time step t, g_t is the gradient at time step t, β_1 and β_2 are the exponential decay rates for the first and second moments, respectively, $\hat{m}_t$ and $\hat{v}_t$ are the bias-corrected first and second moments, α is the learning rate, and ϵ is a small constant used for numerical stability.

2. **Batch Size**: The batch size of 30 which means that the model will be updated after processing 30 samples at a time.

3. **Loss Function**: There is usage of MIL(Multiple Instance Learning) Ranking Loss function proposed in [6] and modified it for better performance. The updated version of MIL is shown via Equation 3.5.

$$
\begin{aligned}
\text{loss} =& \frac{1}{N}\Bigg(\sum_{i=1}^{N} \text{ReLU}\Big(1 - y_{\text{am}} + y_{\text{nm}}\Big) \\
&+ \sum_{i=1}^{N}\Bigg(\sum_{j=1}^{T} y_{\text{anomaly}}, j \times \lambda\Bigg) \\
&+ \sum_{j=1}^{31}\Big(y_{\text{pred}}, i, j - y_{\text{pred}}, i, j+1\Big)^2 \times \lambda\Bigg),
\end{aligned} \tag{3.5}
$$

where N is the batch size, T is the sequence length, y_{am} is the maximum value in the y_{anomaly} tensor, y_{nm} is the maximum value in the y_{normal} tensor, y_{anomaly}, j is the anomaly score for the j-th data point in the batch, y_{pred}, i, j is the predicted value for the j-th time step of the i-th data point in the batch, $y_{\text{pred}}, i, j+1$ is the predicted value for the $j+1$-th time step of the i-th data point in the batch, and λ is a hyperparameter that controls the strength of regularization and for our experiment, it is set to 8×10^{-5}.

4. **Learning Rate**: The project consists of 0.001 as our initial learning rate and get updated at the milestone of 25 and 50 epochs. Usually, the learning rate controls the step size taken during optimization and it determines how quickly the model parameters are updated in response to the estimated error gradient.

5. **Weight Decay**: Weight decay is a regularization technique used to prevent overfitting in models by adding a penalty term to the loss function that discourages large weights.A weight decay of 1×10^{-12}, which means that the penalty term will be proportional to the square of the weights and scaled by this factor

6. **No. of Epochs**: An epoch refers to one full iteration through the entire training dataset. The number of epochs is the number of times the model will see the complete training dataset during the training process. For our approach, the value is set to 100, which means that the model will observe the training data 100 times during the training phase.

Chapter 4

Experimental Analysis

4.1 Environment

4.1.1 Google Colaboratory

This project uses Google Colaboratory as the Integrated Development Environment (IDE) and uses Google's free cloud service, which includes access to the Tesla K80 GPU. The dataset for the project is kept on Google Drive, making it more accessible than local storage. All models are saved on Google Drive for future use during the training process. We used Compute Canada due to the computation restriction for feature extraction.

4.1.2 Compute Canada

Compute Canada is a national organization that provides sophisticated research computing resources and support services to Canadian university academics. It runs a world-class network of high-performance computing (HPC) clusters, storage systems, and software tools to help academics conduct data-intensive and computationally-intensive research. For this project, the Cedar cluster is used for feature extraction. The cluster includes 3600 compute nodes, each with 2 Intel Xeon Gold 6148 processors, 192 GB of RAM, and a variety of NVIDIA Tesla GPUs (P100, V100, and T4).

4.2 Programming Language

For this Deep Learning project, Python was the chosen programming language due to its vast collection of open-source libraries. The project was implemented using Python version 3.0. Following are some of the libraries used in this research.

4.2.1 Pytorch

PyTorch is an open-source machine-learning library used for building and training neural networks. It provides a flexible framework for implementing advanced algorithms and features dynamic computational graphs, automatic differentiation, data parallelism, and support for distributed training. Its features make it a popular choice for researchers and developers alike.

4.2.2 Matplotlib

Matplotlib is a popular open-source Python library used for creating static, animated, and interactive visualizations in 2D and 3D. It provides a range of plotting functionalities, including line plots, scatter plots, bar plots, histograms, and more.

4.2.3 Numpy

NumPy is a Python library used for scientific computing and data analysis that provides an efficient N-dimensional array object for performing mathematical operations on large datasets. It also includes a comprehensive collection of mathematical functions, random number generators, linear algebra routines, and Fourier transforms.

4.2.4 Sklearn

Scikit-learn, commonly abbreviated as sklearn, is a popular machine-learning library that provides a wide range of tools for data mining and analysis, including classification, regression, clustering, and dimensionality reduction, among others. And is built on top of NumPy, SciPy, and matplotlib libraries, and can be used in conjunction with other Python libraries such as pandas for data preprocessing and visualization.

4.3 Evaluation Metrics

The Area Under the Curve (AUC) is a commonly used evaluation metric for binary classification models[1]. As described in Eq. (4.2), it is calculated as the area under the Receiver Operating Characteristic (ROC) curve, which plots true positive rate (TPR) against false positive rate (FPR) for different classification thresholds.

The other way to evaluate a binary classification model using AUC as a metric, which is clearly demonstarted in Eq. (4.1) the first step is to compute the ROC curve using the predicted probabilities and true class labels. Once the ROC curve is obtained, the AUC can be calculated using the integral formula.

$$AUC = \int_{-\infty}^{\infty} ROC(x)dx \tag{4.1}$$

where ROC(x) represents the ROC curve at a given threshold value x, and dx denotes the differential element in the integral. The resulting value ranges from 0.5 to 1.0, where 0.5 indicates a random classification and 1.0 indicates a perfect classification.

The formula for AUC is as follows:

$$AUC = \frac{1}{2}(TPR + TNR - 1), \tag{4.2}$$

where TPR is the True Positive Rate (also known as sensitivity), TNR is the True Negative Rate (also known as specificity).Both of which are described in Eq. (4.3) and Eq. (4.4) respectively.

[1] https://data.library.virginia.edu/roc-curves-and-auc-for-models-used-for-binary-classification/

TPR is calculated as:

$$TPR = \frac{TP}{TP + FN}, \tag{4.3}$$

where TP is the number of true positives and FN is the number of false negatives.

TNR is calculated as:

$$TNR = \frac{TN}{TN + FP}, \tag{4.4}$$

where TN is the number of true negatives and FP is the number of false positives.

Note that the AUC ranges from 0 to 1, with a value of 0.5 indicating random guessing, and a value of 1 indicating perfect classification performance.

The Reason behind selecting AUC as an evaluation metrics for our binary classification is as follows:-

Firstly, AUC is insensitive to class imbalance, which is important because, in many binary classification tasks, the two classes are not equally represented in the dataset. Accuracy can be misleading in such cases because a classifier that always predicts the majority class can achieve a high accuracy despite being useless. AUC considers the relative ranks of predictions, making it a more appropriate metric when classes are imbalanced.

Secondly, AUC is threshold-independent, which means it summarizes the performance of a classifier across all possible threshold choices. Unlike other metrics, such as accuracy or precision, which can vary depending on the threshold choice, AUC is unaffected by the decision threshold and provides a more robust evaluation of classifier performance.

Thirdly, AUC is a ranking-based metric. AUC evaluates the ability of a classifier to rank instances according to their true class labels. This means that AUC is less sensitive to outliers or noise in the dataset than other metrics such as accuracy or precision. As long as the classifier ranks the positive instances higher than the negative instances on average, it will achieve a higher AUC.

Finally, The AUC value ranges from 0 to 1 and a higher value indicates better classification performance. It can be interpreted as the probability that a randomly chosen positive instance will be ranked higher than a randomly chosen negative instance by the classifier. This makes it an easy and intuitive metric for comparing different classifiers or model configurations.

To summarize, the AUC is a widely used measure for assessing the performance of binary classification models. One of its key advantages is that it is unaffected by imbalances in class representation or threshold selection. In addition, it is based on relative ranking and provides a clear indication of how well a classifier is performing. As a result, the AUC is a reliable and valuable metric for evaluating binary classification models, particularly in cases where other evaluation metrics may be susceptible to common problems encountered in such problems.

4.4 Performance Analysis

4.4.1 Quantitative Analysis

Table 4.1: Performance Comparison of Different Methods

Method	**Features**	**AUC(%)**	**% Gain**
Sultani et al. [6]	C3D-RGB	75.41	↓2.51
Sultani et al. [6] [**Baseline**]	I3D-RGB	77.92	-
RTFM [62]	I3D-RGB	84.30	↑6.38
MGFN [63]	I3D-RGB	86.98	↑9.06
Ours	**I3D-RGB + I3D-OF**	**87.52**	**↑9.6**

As shown in Table 4.1, Comparision of the approach with three different approaches, of which [6] is taken as our baseline method, where the UCF-Crime Dataset is originally proposed. The other approaches are taken into consideration are [62], which proposed an improvement on [6] and [63], which shows the state of the art performance on UCF-Crime Dataset.

The main difference between our approach and other approaches is the way we extract our features to feed into our model. Most of the current and past approaches have utilized single-stream Models to extract their features. They have only utilized the RGB Frames of the videos, whereas we have also integrated the optical flow information, which is a representation of how the pixels in the video are moving over time. By using both the RGB and optical flow streams, the two-stream model is able to capture both appearance and motion information in the video, which leads to improved performance for our approach.

Our Baseline approach [6] has utilized both C3D and I3D to extract features, given only the RGB frames of the videos. It was observed that I3D's performance is better than C3D's performance by 2.5%, so we have decided to go with the I3D model for our feature extraction task but with the single stream approach we only achieved 84.02% AUC, so we decided to go with two stream approach with I3D model where both RGB frames and optical flow information is utilized to extract better features. Usually, this approach is only been used for the action recognition task [64], where we have modified it to extract features from the videos. Also, we have made a few modifications to the MIL(Multiple Instance Learning) Loss function to get better separability between abnormal and normal instances. Due to all these modifications, we have achieved a performance of 87.52%, which is a 9.6% gain.

In the project the RTFM approach [62], which is a technique that enhances the robustness of the MIL [6] approach in recognizing positive instances in abnormal videos. It achieves this by training a function to learn feature magnitudes effectively. The method utilizes dilated convolutions and self-attention mechanisms to capture both long- and short-term temporal dependencies for accurate feature magnitude learning, we tried to incorporate self-attention mechanisms for our approach, where we got only 61.67% AUC.

Comparing it with the state-of-the-art approach, MGFN [63], which employs a contrastive learning approach that focuses on learning the differences between normal and abnormal events in videos without the need for precise labelling for each frame. The model accomplishes this by employing a ”glance-and-focus” mechanism to selectively attend to areas of the video that are more likely to contain anomalies. Inspired by this, we have done predictions using only the temporal annotations available for testing videos, whereas training is done using video-level labels.”Glance and Focus” mechanism is complex in terms of its execution, so to mitigate it we utilized better features using two stream I3D feature extractor and feed it to 4 layer fully connected neural networks, which is simple in nature and gave us the performance of 87.52%, which 0.54% improvement.

4.4.2 Qualitative Analysis

The training progress for the proposed solution on the UCF-Crime dataset is shown in Fig. 4.1. It shows the training loss progression with each epoch, which is computed using the modified version of the MIL Ranking loss function. It is clearly observed that the magnitude of loss is drastically decreasing till the 30^{th} epoch after that it remains stable, which few ups and down till the last epoch. During the training process, to avoid the issue of overfitting, early stopping regularization[2] is employed with patience of 10 and minimum delta value of 0.001, with training loss set as a criteria.

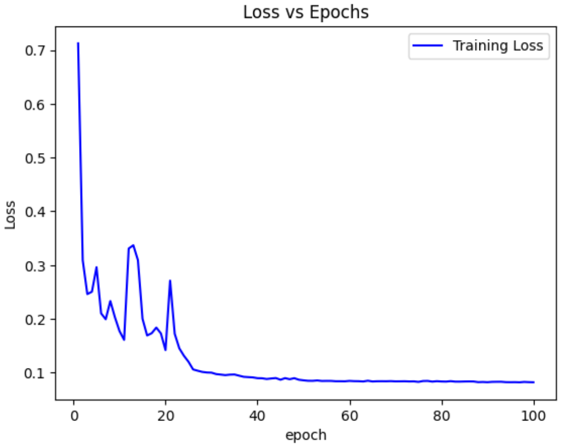


Figure 4.1: Training Progress

The testing progression is clearly visible in Fig. 4.2. Here along with loss progress,

[2] https://pytorch.org/ignite/generated/ignite.handlers.early_stopping.EarlyStopping.html

AUC progression with each epoch is also displayed.

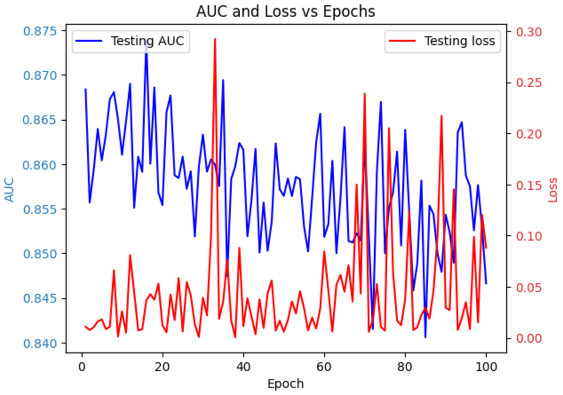
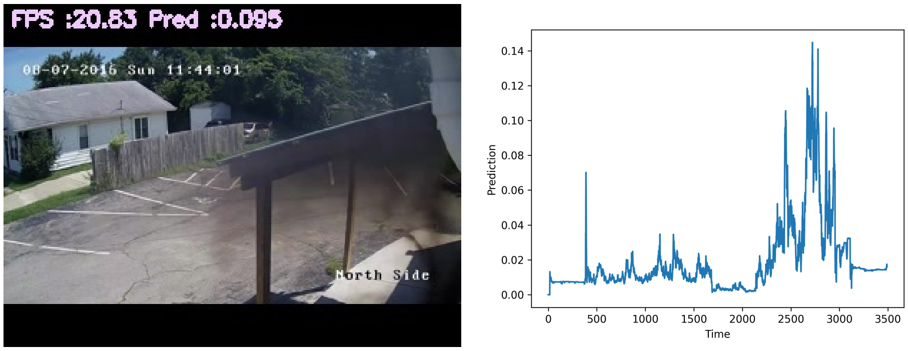


Figure 4.2: Testing Progress

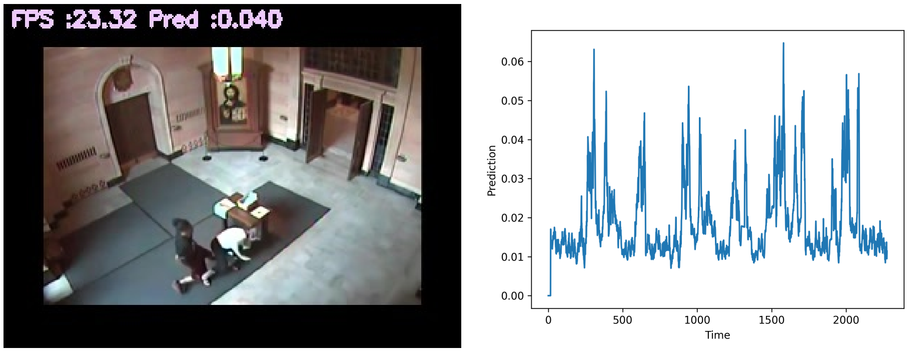
Fig. 4.3a, 4.4a and 4.5a shows the video level snippets from arson, abuse and robbery video set from 13 different anomalies video set present in the UCF-Crime dataset. It also contains the FPS value and real-time prediction score for each frame. While, Fig. 4.3b, 4.4b and 4.5b shows temporal analysis of anomaly prediction on videos used in Fig. 4.3a, 4.4a and 4.5a respectively. Here higher the prediction value, the higher the chances of that event being anomalous.



(a) Video snippet from arson anomaly set

(b) Temporal Analysis of Anomaly Prediction on 4.3a

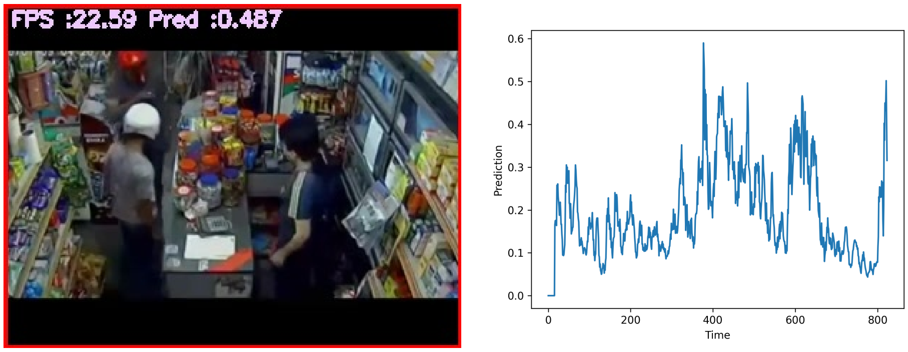
Figure 4.3: Anomaly Detection Analysis on Arson Video Set



(a) Video snippet from abuse anomaly set

(b) Temporal Analysis of Anomaly Prediction on 4.4a

Figure 4.4: Anomaly Detection Analysis on Abuse Video Set



(a) Video snippet from robbery anomaly set

(b) Temporal Analysis of Anomaly Prediction on 4.5a

Figure 4.5: Anomaly Detection Analysis on Robbery Video Set

Chapter 5

Discussion and Conclusion

5.0.1 Discussion

A discussion regarding all the Chapters is presented in this chapter. Some conclusions are also drawn, as well as recommendations for future research.

Chapter 2 discusses various existing approaches to solve the problem on different datasets. We divided the problem solution into 7 different Learning methods, which are Transfer learning-based models, Ensemble learning based models, Continual learning-based models, Representation learning-based models, Reinforcement learning-based models, GNN based models, and Generative models. It was deduced that major research was done on Transfer Learning based models and Generative models. Also, most of the existing research is only focused on developing a complex model that can detect anomalies from the videos, while ignoring the fact that extracting features from the given video frames is also as important. Also, Most of the existing approach is only focused on extracting features from the RGB Frames, while overlooking the optical information of the given frames, which can help the model in better prediction of the anomalies. Hence, after careful consideration of the current methods, this project utilized two stream I3D for feature extraction, which resulted in better performance than its contemporary. It also discusses various benchmark datasets available for video anomaly detection, from all the available datasets, UCF-Crime is chosen for this project, due to its complexity and the number of videos available.

Chapter 3 discusses the proposed solution of the video anomaly detection task on the UCF-Crime dataset. The Recommended approach is divided into 3 Phases where Phase 1 is focused on gathering input videos and extracting segments from both anomalous videos and normal videos, here 32 segments are taken into consideration. Phase 2 is all about extracting features from the provided segments. To begin with, firstly, pre-processing is performed which includes frame resizing, center, and random cropping, normalization, RGB to BGR conversion, optical flow computation, and its normalization. Phase 3 includes the classification subnetwork, where extracted features are provided to the 4 layered fully connected neural networks, and a modified version of MIL is used as a loss function for anomaly prediction.

Chapter 4 discusses the experimental study where discourse regarding various experiments conducted during the course of the project, comparison with the baseline, and state-of-the-art approach, Qualitative and Quantitative analysis of the result along with the discussion regarding the project's environment is presented.

5.0.2 Conclusion

The task of video anomaly detection remains an ongoing challenge, requiring existing research to achieve a better surveillance system. Existing approaches have made significant strides in Transfer Learning and Generative models, with features playing a crucial role in improving classification. However, most approaches only consider RGB frames and do not take advantage of valuable optical flow information, which could further enhance performance. In this study, the method proposed uses the two streams of the I3D model as feature extractors, combined with a basic 5-layer fully connected neural network for classification. Also, modifications have been made to the existing MIL ranking loss function to achieve a performance improvement of 12.11% over the Baseline [6] and 0.54% over the state of the art [63]. Overall, the work highlights the potential of deep feature-based anomaly detection for video surveillance and provides a promising direction for future research in this field.

5.0.3 Future Work

Below are a couple of suggestions for future improvements, based on the previous discussion:

- *Experiment with various datasets*: The model can be tested on various datasets to see how it performs on various types of videos, such as indoor and outdoor surveillance footage. This may assist in identifying the model's strengths and weaknesses and fine-tuning it accordingly.
- *Increase the number of anomalies in the dataset*: The current dataset may not cover all of the possible anomalies in surveillance videos. Increasing the

number of anomalies in the dataset can assist the model in learning to detect a broader range of abnormal events.

- *Investigate alternative methodologies*: While the deep feature-based approach has yielded promising results, other methodologies can be investigated to achieve even better results. For example, unsupervised learning algorithms such as Graph Attention Networks (GAT) can be used to detect anomalies.

- *Improve the accuracy of the model*: Improve the model's accuracy: The model's accuracy can be improved further by fine-tuning the hyperparameters and optimising the neural network architecture. This can help reduce false positives and increase the model's sensitivity to anomalous events.

- *Develop a real-time anomaly detection system*: The current model is offline and requires video data pre-processing before anomaly detection. Creating a real-time system that can process and analyze video data in real-time is a practical application of the model.

- *Extend the application of the model*: The technique can be used to areas other than video surveillance, like traffic monitoring, fraud detection, and medical imaging. Using deep feature-based techniques, this can aid in tackling the issues associated with anomaly identification in these areas.

  Overall, the future work for your project involves fine-tuning the existing model, exploring new methodologies, and extending its application to other domains. These efforts can help improve the accuracy and practicality of deep feature-based anomaly detection for video surveillance.

Appendices

Appendix A

IEEE Permission to Reprint

In reference to IEEE copyrighted material, which is used with permission in this thesis, the IEEE does not endorse any of Lakehead University's products or services. International or personal use of the material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to IEEE

Appendix B

Code Samples

The code samples and documentation are available at the Link and Repository

Or you are welcome to reach out to the authors for any concerns at dshah33@lakeheadu.ca and bhattp@lakheadu.ca.

Appendix C

Members Contribution

Table C.1: % Members Contribution

Task	**Parth Bhatt**	**Dhruva Shah**
Literature Review	50	50
Model Development	50	50
Experiments	50	50
Report Writing	60	40
Conference Paper	40	60

Bibliography

[1] Joey Tianyi Zhou, Jiawei Du, Hongyuan Zhu, Xi Peng, Yong Liu, and Rick Siow Mong Goh. Anomalynet: An anomaly detection network for video surveillance. *IEEE Transactions on Information Forensics and Security*, 14(10):2537–2550, 2019.

[2] Xing Hu, Shiqiang Hu, Yingping Huang, Huanlong Zhang, and Hanbing Wu. Video anomaly detection using deep incremental slow feature analysis network. *IET Computer Vision*, 10(4):258–267, 2016.

[3] Jia-Xing Zhong, Nannan Li, Weijie Kong, Shan Liu, Thomas H Li, and Ge Li. Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1237–1246, 2019.

[4] Javier Sánchez Pérez, Enric Meinhardt-Llopis, and Gabriele Facciolo. Tv-l1 optical flow estimation, Jul 2013.

[5] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset, 2018.

[6] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6479–6488, 2018.

[7] Jessie James P Suarez and Prospero C Naval Jr. A survey on deep learning techniques for video anomaly detection. *arXiv preprint arXiv:2009.14146*, 2020.

[8] Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019.

[9] Rashmiranjan Nayak, Umesh Chandra Pati, and Santos Kumar Das. A comprehensive review on deep learning-based methods for video anomaly detection. *Image and Vision Computing*, 106:104078, 2021.

[10] John G Munyua, Geoffrey M Wambugu, and Stephen T Njenga. A survey of deep learning solutions for anomaly detection in surveillance videos. 2021.

[11] Sijie Zhu, Chen Chen, and Waqas Sultani. Video anomaly detection for smart surveillance. In *Computer Vision: A Reference Guide*, pages 1–8. Springer, 2020.

[12] B Ravi Kiran, Dilip Mathew Thomas, and Ranjith Parakkal. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. *Journal of Imaging*, 4(2):36, 2018.

[13] Bharathkumar Ramachandra, Michael Jones, and Ranga Raju Vatsavai. A survey of single-scene video anomaly detection. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[14] S Geetha, CS Abhishek, and CS Akshayanat. Machine vision based fire detection techniques: a survey. *Fire Technology*, 57(2):591–623, 2021.

[15] S Anoopa and A Salim. Survey on anomaly detection in surveillance videos. *Materials Today: Proceedings*, 2022.

[16] Rajesh Kumar Yadav and Rajiv Kumar. A survey on video anomaly detection. In *2022 IEEE Delhi Section Conference (DELCON)*, pages 1–5. IEEE, 2022.

[17] Mohammad Sabokrou, Mahmood Fathy, and Mojtaba Hoseini. Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder. *Electronics Letters*, 52(13):1122–1124, 2016.

[18] Shuheng Lin, Hua Yang, Xianchao Tang, Tianqi Shi, and Lin Chen. Social mil: Interaction-aware for crowd anomaly detection. In *2019 16th IEEE international conference on advanced video and signal based surveillance (AVSS)*, pages 1–8. IEEE, 2019.

[19] Fernando P dos Santos, Leonardo SF Ribeiro, and Moacir A Ponti. Generalization of feature embeddings transferred from different video anomaly detection domains. *Journal of Visual Communication and Image Representation*, 60:407–416, 2019.

[20] Suprit Bansod and Abhijeet Nandedkar. Transfer learning for video anomaly detection. *Journal of Intelligent & Fuzzy Systems*, 36(3):1967–1975, 2019.

[21] Lucas Pinheiro Cinelli. Anomaly detection in surveillance videos using deep residual networks. *Universidade Federal do Rio de Janeiro, Rio de Janeiro*, 2017.

[22] Tiago S Nazare, Rodrigo F de Mello, and Moacir A Ponti. Are pre-trained cnns good feature extractors for anomaly detection in surveillance videos? *arXiv preprint arXiv:1811.08495*, 2018.

[23] Waseem Ullah, Amin Ullah, Ijaz Ul Haq, Khan Muhammad, Muhammad Sajjad, and Sung Wook Baik. Cnn features with bi-directional lstm for real-time anomaly detection in surveillance networks. *Multimedia Tools and Applications*, 80(11):16979–16995, 2021.

[24] Ali Khaleghi and Mohammad Shahram Moin. Improved anomaly detection in surveillance videos based on a deep learning method. In *2018 8th Conference of AI & Robotics and 10th RoboCup Iranopen International Symposium (IRANOPEN)*, pages 73–81. IEEE, 2018.

[25] Kun Liu, Minzhi Zhu, Huiyuan Fu, Huadong Ma, and Tat-Seng Chua. Enhancing anomaly detection in surveillance videos with transfer learning from action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4664–4668, 2020.

[26] Sagar Chhetri, Abeer Alsadoon, Thair Al-Dala'in, PWC Prasad, Tarik A Rashid, and Angelika Maag. Deep learning for vision-based fall detection system: Enhanced optical dynamic flow. *Computational Intelligence*, 37(1):578–595, 2021.

[27] Federico Landi, Cees GM Snoek, and Rita Cucchiara. Anomaly locality in video surveillance. *arXiv preprint arXiv:1901.10364*, 2019.

[28] Yi Zhu and Shawn Newsam. Motion-aware feature for improved video anomaly detection. *arXiv preprint arXiv:1907.10211*, 2019.

[29] Tuan-Hung Vu, Jacques Boonaert, Sebastien Ambellouis, and Abdelmalik Taleb-Ahmed. Multi-channel generative framework and supervised learning for anomaly detection in surveillance videos. *Sensors*, 21(9):3179, 2021.

[30] Yumna Zahid, Muhammad Atif Tahir, and Muhammad Nouman Durrani. Ensemble learning using bagging and inception-v3 for anomaly detection in surveillance videos. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 588–592. IEEE, 2020.

[31] M Murugesan and S Thilagamani. Efficient anomaly detection in surveillance videos based on multi layer perception recurrent neural network. *Microprocessors and Microsystems*, 79:103303, 2020.

[32] Ke Xu, Tanfeng Sun, and Xinghao Jiang. Video anomaly detection and localization based on an adaptive intra-frame classification network. *IEEE Transactions on Multimedia*, 22(2):394–406, 2019.

[33] Keval Doshi and Yasin Yilmaz. Continual learning for anomaly detection in surveillance videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 254–255, 2020.

[34] Behnam Sabzalian, Hossein Marvi, and Alireza Ahmadyfard. Deep and sparse features for anomaly detection and localization in video. In *2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA)*, pages 173–178. IEEE, 2019.

[35] Sabrina Aberkane and Mohamed Elarbi. Deep reinforcement learning for real-world anomaly detection in surveillance videos. In *2019 6th International Conference on Image and Signal Processing and their Applications (ISPA)*, pages 1–5. IEEE, 2019.

[36] K Kavikuil and J Amudha. Leveraging deep learning for anomaly detection in video surveillance. In *First International Conference on Artificial Intelligence and Cognitive Computing*, pages 239–247. Springer, 2019.

[37] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 733–742, 2016.

[38] Jefferson Ryan Medel and Andreas Savakis. Anomaly detection in video using predictive convolutional long short-term memory networks. *arXiv preprint arXiv:1612.00390*, 2016.

[39] Manassés Ribeiro, André Eugênio Lazzaretti, and Heitor Silvério Lopes. A study of deep convolutional auto-encoders for anomaly detection in videos. *Pattern Recognition Letters*, 105:13–22, 2018.

[40] Radu Tudor Ionescu, Fahad Shahbaz Khan, Mariana-Iuliana Georgescu, and Ling Shao. Object-centric auto-encoders and dummy anomalies for abnormal event detection in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7842–7851, 2019.

[41] Yong Shean Chong and Yong Haur Tay. Abnormal event detection in videos using spatiotemporal autoencoder. In *International symposium on neural networks*, pages 189–196. Springer, 2017.

[42] Elvan Duman and Osman Ayhan Erdem. Anomaly detection in videos using optical flow and convolutional autoencoder. *IEEE Access*, 7:183914–183923, 2019.

[43] Anitha Ramchandran and Arun Kumar Sangaiah. Unsupervised deep learning system for local anomaly event detection in crowded scenes. *Multimedia Tools and Applications*, 79(47):35275–35295, 2020.

[44] Sukalyan Bhakat and Ganesh Ramakrishnan. Anomaly detection in surveillance videos. In *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, pages 252–255, 2019.

[45] Karishma Pawar and Vahida Attar. Application of deep learning for crowd anomaly detection from surveillance videos. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 506–511. IEEE, 2021.

[46] Khac-Tuan Nguyen, Dat-Thanh Dinh, Minh N Do, and Minh-Triet Tran. Anomaly detection in traffic surveillance videos with gan-based future frame prediction. In *Proceedings of the 2020 International Conference on Multimedia Retrieval*, pages 457–463, 2020.

[47] Medhini G Narasimhan et al. Dynamic video anomaly detection and localization using sparse denoising autoencoders. *Multimedia Tools and Applications*, 77(11):13173–13195, 2018.

[48] Mohammad Sabokrou, Mohsen Fayyaz, Mahmood Fathy, Zahra Moayed, and Reinhard Klette. Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes. *Computer Vision and Image Understanding*, 172:88–97, 2018.

[49] Yiru Zhao, Bing Deng, Chen Shen, Yao Liu, Hongtao Lu, and Xian-Sheng Hua. Spatio-temporal autoencoder for video anomaly detection. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1933–1941, 2017.

[50] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1705–1714, 2019.

[51] Yaxiang Fan, Gongjian Wen, Deren Li, Shaohua Qiu, Martin D Levine, and Fei Xiao. Video anomaly detection and localization via gaussian mixture fully convolutional variational autoencoder. *Computer Vision and Image Understanding*, 195:102920, 2020.

[52] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. Future frame prediction for anomaly detection–a new baseline. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6536–6545, 2018.

[53] Nasaruddin Nasaruddin, Kahlil Muchtar, Afdhal Afdhal, and Alvin Prayuda Juniarta Dwiyantoro. Deep anomaly detection through visual attention in surveillance videos. *Journal of Big Data*, 7(1):1–17, 2020.

[54] Keval Doshi and Yasin Yilmaz. Online anomaly detection in surveillance videos with asymptotic bound on false alarm rate. *Pattern Recognition*, 114:107865, 2021.

[55] Mohammad Sabokrou, Mohsen Fayyaz, Mahmood Fathy, and Reinhard Klette. Deep-cascade: Cascading 3d deep neural networks for fast anomaly detection

and localization in crowded scenes. *IEEE Transactions on Image Processing*, 26(4):1992–2004, 2017.

[56] Keval Doshi and Yasin Yilmaz. Any-shot sequential anomaly detection in surveillance videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 934–935, 2020.

[57] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 fps in matlab. In *Proceedings of the IEEE international conference on computer vision*, pages 2720–2727, 2013.

[58] Jhih-Ciang Wu, He-Yen Hsieh, Ding-Jie Chen, Chiou-Shann Fuh, and Tyng-Luh Liu. Self-supervised sparse representation for video anomaly detection. *Lecture Notes in Computer Science*, page 729–745, 2022.

[59] Neelu Madan, Nicolae-Catalin Ristea, Radu Tudor Ionescu, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moeslund, and Mubarak Shah. Self-supervised masked convolutional transformer block for anomaly detection. *arXiv.org*, 2022.

[60] Yu Tian, Guansong Pang, Yuanhong Chen, Rajvinder Singh, Johan W Verjans, and Gustavo Carneiro. Weakly-supervised video anomaly detection with robust temporal feature magnitude learning. *arXiv.org*, 2021.

[61] Jun-Hyung Yu, Jeong-Hyeon Moon, and Kyung-Ah Sohn. Attention-guided residual frame learning for video anomaly detection. *Multimedia Tools and Applications*, Sep 2022.

[62] Yu Tian, Guansong Pang, Yuanhong Chen, Rajvinder Singh, Johan Verjans, and Gustavo Carneiro. *Weakly-supervised Video Anomaly Detection with Robust Temporal Feature Magnitude Learning.*

[63] Yingxian Chen, Zhengzhe Liu, Baoheng Zhang, Wilton Fok, Xiaojuan Qi, and Yik-Chung Wu. *MGFN : Magnitude-Contrastive Glance-and-Focus Network for Weakly-Supervised Video Anomaly Detection.*

[64] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset, 2017.