

Context-Aware Zero-Shot Anomaly Detection in Surveillance Using Contrastive and Predictive Spatiotemporal Modeling

By

Md. Abrar Hasan
23241115

Md. Rashid Shahriar Khan
21201585

Mohammod Tareq Aziz Justice
20101557

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
July 2025

© 2025. BRAC University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at BRAC University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



Md. Abrar Hasan
23241115



Md. Rashid Shahriar Khan
21201585



Mohammod Tareq Aziz Justice
20101557

Approval

The thesis titled “Context-Aware Zero-Shot Anomaly Detection in Surveillance Using Contrastive and Predictive Spatiotemporal Modeling” submitted by

1. Md. Abrar Hasan (23241115)
2. Md. Rashid Shahriar Khan (21201585)
3. Mohammod Tareq Aziz Justice (20101557)

Of Spring, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on July 22, 2025.

Examining Committee:

Supervisor:
(Member)



Dr. Md. Ashraful Alam

Associate Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)



Md. Tanzim Reza

Senior Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi

Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

Tackling anomalies through surveillance feeds is challenging due to the unpredictable nature of anomalies and their strong dependence on context. Modern video anomaly detection architectures have been shown to thrive in such conditions. Their ability to adapt to intricate and complex patterns serves as the foundation of anomaly detection, especially for unseen scenarios, making the impossible seem tangible. The research demonstrates a novel context-aware zero-shot anomaly detection framework that learns normal spatiotemporal patterns and identifies anomalies without any explicit anomaly examples during training. In order to perform the approach, it proposed a hybrid model which is a combination of TimeSformer, DPC, and CLIP. A TimeSformer-based vision transformer backbone is employed to encode video sequences, capturing rich spatial-temporal features. We integrate Data Predictive Control (DPC) to forecast future video dynamics and flag deviations. Simultaneously, we leverage the vision-language power of CLIP in a semantic stream where the model is conditioned on contextual information and uses text prompts to detect concept-level irregularities in a zero-shot fashion. These components are jointly optimized using InfoNCE and Contrastive Predictive Coding (CPC) losses, enabling the model to align video inputs with their semantic and temporal contexts without ever being exposed to anomaly labels. To condition decisions on the situational context, we propose a context-gating mechanism that modulates temporal predictions based on scene-specific text or global video features. During inference, anomalies are flagged based on a fusion of context misalignment and predictive failure, allowing the system to generalize to previously unseen behaviours. Evaluations of our lightweight and fully zero-shot approach achieve a ROC-AUC of 84.5%, and a PR-AUC of 72.3%. This work advances the gap between semantic understanding and temporal prediction in surveillance, laying the foundation for context-sensitive, zero-shot detection systems deployable in dynamic real-world environments.

Keywords: Context Awareness; Zero-Shot Learning; Video Surveillance; Contrastive Learning; Predictive Modeling; Spatiotemporal; TimeSformer

Dedication

All thanks to Almighty Allah, the most magnificent; without His will, this task wouldn't be completed.

It is our great pleasure to see that each member has shown the utmost dedication to our work and all blessings to them.

We shall forever be grateful to our parents for their steadfast confidence in our abilities and assistance, as well as their collaboration, which inspired us to achieve scholastic success. Their heartfelt prayers and uplifting remarks contributed to our continued growth and maturation into the individuals we are today. As we embark on the subsequent phase of our existence, we recognize and appreciate the immense value of their selflessness. We express profound gratitude for their constant support.

Acknowledgement

Allah is the greatest planner. His favour was important to let us finish our thesis. His guidance provided us with direction, strength, and knowledge to overcome obstacles along the way.

We have gotten constant support from our supervisor, Dr. Md. Ashraful Alam. His supervision has made our academic journey smoother and helpful. We want to thank him from the bottom of our hearts. He is not only our great supervisor but also a mentor and an inspiration. The knowledge, constructive criticism, and support that he provided to us have been valuable in helping us shape our study. It also helped us to improve our knowledge of the field. We are really grateful to have had the chance to work under his direction. His commitment to academic quality has been clear in every sector. Also, we could not do it without recognising our co-supervisor, Md. Tanzim Reza sir's continuous availability and desire to share his depth of knowledge. His assistance helped us a lot. The understanding we have received from him will influence our future endeavours.

We are very honoured to have our supervisor and co-supervisor for their guidance and significant influence on our academic and personal well-being.

Table of Contents

Declaration	i
Approval	ii
Abstract	iv
Dedication	v
Acknowledgment	vi
Table of Contents	vii
List of Figures	viii
List of Tables	ix
Nomenclature	x
1 Introduction	1
1.1 Background of the Study	1
1.2 Problem Statement	2
1.3 Research Objectives	3
2 Literature Review	5
3 Proposed Methodology	11
3.1 Methodology Overview	11
3.2 Proposed Model Architecture	12
3.3 Dataset Description	20
3.3.1 Data Handling	20
3.3.2 Exploratory Data Analysis	21
4 Result Analysis	23
4.1 Our Contributions	23
4.2 Results and Comparisons	23
4.3 Ablation Study	27
5 Conclusion	29
Bibliography	30

List of Figures

3.1	Video Architecture Pipeline	14
3.2	Text Pipeline	16
3.3	Joint Training on Prediction Loss & infoNCE	17
3.4	Decoding Raw Pixel Data	21
3.5	Temporal Frame Gaps Analysis	22
3.6	Frame-to-Frame Difference	22
4.1	Performance Comparison on UCF-Crime Dataset	24
4.2	Comparison of mAP and Detection Delays for Anomaly Detection Models	25
4.3	F1-score Comparison of Anomaly Detection Models	26

List of Tables

4.1	Comparison of ROC-AUC & PR-AUC score for Anomaly Detection Models with Zero-Shot Learning on UCF-Crime dataset	24
4.2	Comparison of mAP(%) & Detection Delay(s) for Anomaly Detection Models (with and without Zero-Shot Learning)	25
4.3	Comparison of F1-score for Anomaly Detection Models (with and without Zero-Shot Learning)	26
4.4	Ablation on UCF-Crime (each row toggles a single factor; training budget and optimiser settings are otherwise identical)	27

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AUC Area Under the Curve

CLIP Contrastive Language-Image Pretraining

CNN Convolutional Neural Network

CPC Contrastive Predictive Coding

GRU Gated Recurrent Unit

MLP Multi-Layer Perceptron

NCE Noise Contrastive Estimation

ROC Receiver Operating Characteristic

UCF University of Central Florida

VAD Video Anomaly Detection

ViT Vision Transformer

XD Cross-Domain

YOLOv8 You Only Look Once version 8

Chapter 1

Introduction

1.1 Background of the Study

Anomaly detection is a fundamental aspect of various domains, including finance, healthcare, cybersecurity, etc., where identifying irregular patterns is crucial for ensuring security and operational efficiency. In the realm of surveillance in crowd areas, anomaly detection is particularly important for identifying potential threats. Since surveillance activity is rapidly growing in the world day by day, it has become a crucial demand to analyze videos through automated systems. As a result, video anomaly detection (VAD) has emerged as an important research area that leverages computational models to automate the identification of abnormal events in security videos [10]. Ensuring public safety in environments such as streets, campuses, airports, and shopping malls heavily relies on effective VAD systems that can alert authorities to crimes, accidents, or other irregular behaviors as they occur.

In the early stage, when the video anomaly detection system was newly introduced to us, it used to focus on manually crafted features and statistical models. Traditional computer vision algorithms used motion trajectories, object speed, or pixel-level changes to characterize normal patterns, raising alerts when deviations occur. However, these old systems had limitations as they were hand-crafted. For example, capturing complex spatiotemporal patterns in crowded or dynamic scenes was a major challenge where these approaches were not suitable. After that, a revolutionized anomaly detection technique called deep learning emerged, with the capability of learning from rich feature representations directly from data. These deep learning models automatically learn what defines normal behavior in a video, making them suitable for detecting subtle or complex anomalies that traditional algorithms often miss [28].

A major challenge of anomaly detection in surveillance systems is the lack of anomalous examples for training. Because anomaly events occur rarely in our real life world. That's why it becomes hard to gather extensive training data for every possible anomaly incident [5]. As a result, many systems adopt an unsupervised or semi-supervised approach where models are trained only on normal video segments, learning a model of normality, and then flagging deviations without having seen examples of anomalies in training. This thing aligns with a zero-shot anomaly detection scenario, where the system must detect novel abnormal events without explicit

training on those events. The use of deep autoencoders and predictive models has been particularly effective in this context. For instance, a model might be trained to reconstruct video frames or predict future frames, and large prediction errors indicate an anomaly. Such methods have achieved success in detecting anomalies like unusual object motions or the sudden appearance of new objects in surveillance scenes [24].

While deep learning has advanced video anomaly detection systems, a new frontier is incorporating context-awareness into anomaly detection. Context can include the location, time, or prevailing conditions under which video is captured. In real-world surveillance, whether an activity is anomalous often depends on contextual factors. For example, seeing a person walking on a school campus at noon on a weekday is normal, but the same activity at midnight might be highly suspicious. The definition of normal behavior is context-dependent [32]. Traditional anomaly detectors that ignore such context may either miss anomalies or raise false alarms in contextually permissible situations. This insight has encouraged research into context-aware models that can adjust their notion of normality based on contextual inputs like day of week or location type metadata. Another emerging trend is leveraging vision-language models that fuse visual and textual information. Models such as CLIP, which align images and natural language, present an opportunity to introduce high-level semantic knowledge into anomaly detection. By describing expected scenarios in words and matching them with visual input, such models hold promise for true zero-shot anomaly recognition, where even complex events can be detected by understanding their semantic context [31].

In summary, the background of this study lies at the intersection of surveillance anomaly detection, context-aware modeling, and zero-shot learning. Recent advances provide building blocks like TimeSformer, a transformer for video understanding [3] for capturing spatiotemporal features, Contrastive Predictive Coding (CPC) for self-supervised sequence learning [2], and CLIP for linking visual data with textual context [4]. These tools enable a new class of surveillance systems that can learn normal patterns, account for context, and generalize to detect previously unseen anomalies. This thesis explores such a system, aiming to advance anomaly detection capabilities in surveillance through contrastive and predictive spatiotemporal modeling with context-awareness.

1.2 Problem Statement

Despite the fact that video anomaly detection systems are improving day by day, these cannot reaching up to the mark in terms of efficiency. Basically, several critical challenges remain unaddressed in existing approaches. A major limitation is the current detection algorithms are suffering from the lack of context awareness as well as zero-shot generalization. They often treat all environments and times homogenously, implicitly assuming that anomalousness is a universal thing. But we know that this is not true because in the real-life world, a normal behaviour from any particular scene can vary dramatically with context, such as time-of-day or the presence of special events. Current systems struggle with the long-term surveillance

settings where cameras operate continuously and contexts are constantly changing. For example, a large crowd gathering might be a normal scene at a particular time, but it becomes an anomaly at another time. Standard anomaly detectors without context conditioning would either miss the event in the former case or falsely alarm in the latter. This gap between academic benchmarks and real-world deployment means that many algorithms are not immediately applicable to practical surveillance networks [32].

Another side of the problem is the failure to detect novel anomalies by zero-shot technique. Most deep learning-based methods, even if unsupervised, are tuned to detect anomalies similar to those implicitly present in the normal training data distribution. These cause an inability to detect new types of anomalies. Also, this often results in poor performance when encountering anomalies that are subtle or contextually complex. For example, if a model has never been trained with the scene of raining, then a traditional model might flag the scene of a person carrying an umbrella as an anomaly or might ignore it altogether, depending on how it learned normal patterns. The imbalanced nature of abnormal incidents makes it infeasible to predefine all anomalies in a training set. This is essentially a zero-shot learning problem within anomaly detection, where the system failed to identify or categorize an anomaly that was never explicitly labeled during training.

Existing approaches have only partially addressed these issues. Some supervised methods include high-level labels, but they still cannot push external knowledge and struggle to localize anomalies in time. Current researches try to adapt vision-language models like CLIP for anomaly detection for those behaviors that are weakly supervised, but these are generally limited to scenarios where a pre-defined list of anomaly classes is provided. Some approaches use semantic knowledge to classify anomalies into types, but still these methods typically require fine-tuning or at least knowledge of the anomaly classes, which actually doesn't fall under the zero-shot technique. Moreover, these models can detect what is anomaly but can't say when a scene can be recognized as an anomaly.

However, considering all the things above, the central problem addressed by our thesis is no fully capable surveillance anomaly detection model which can detect abnormal behaviours using both context-aware and zero-shot approach.

1.3 Research Objectives

The primary objective of this research is to design a novel deep learning model for surveillance videos that incorporates context-conditioning. The model should learn representations of normal behavior not in isolation, but in relation to contextual factors. This involves creating a dual-stream architecture where one stream models the video's spatiotemporal content and another stream provides contextual input, enabling the system to adjust its understanding of normality based on context.

Secondly, leveraging contrastive learning and external semantic knowledge to allow the detection of anomalies that were never explicitly observed during training. Specifically, integrate a vision-language model like CLIP to imbue the system with

semantic understanding, so that if an anomalous event corresponds to a known concept, the model can recognize it via textual descriptions even without direct training examples. Additionally, use self-supervised predictive modeling to learn generalizable temporal patterns that can flag novel deviations.

Thirdly, utilizing a TimeSformer (Time-based Transformer) to capture spatial and temporal features of video sequences in a unified framework. Also, employing a contrastive predictive coding loss to train the model to predict future frame representations, reinforcing the learning of normal spatiotemporal dynamics. By combining contrastive learning (for context and semantic alignment) with predictive modeling (for temporal consistency), to ensure that the learned representation is robust to variations and sensitive to meaningful deviations.

Finally, evaluating the developed model on both public benchmark datasets and a context-rich surveillance dataset rigorously. Key performance indicators include frame-level and video-level anomaly detection accuracy (e.g., measured by area under the ROC curve), as well as the false alarm rate in various contexts. So, our objective is to compare the performance against state-of-the-art anomaly detection methods (both context-agnostic and context-aware variants) to demonstrate improvements. Performing statistical analyses to verify the significance of performance gains and to quantify the contributions of each component (context conditioning, contrastive loss, predictive coding, etc.) to the overall system.

Achieving these objectives will result in a context-aware, zero-shot anomaly detection system that advances the field of intelligent surveillance and has practical implications for deploying artificial intelligence in real-world security environments.

Chapter 2

Literature Review

Early approaches to surveillance anomaly detection were often based on manually crafted features and assumptions about typical motion patterns. For example, techniques like trajectory analysis and optical flow were widely used to model normal movement within a scene, with any deviations from these patterns considered anomalies [27]. Traditional methods often relied on statistical models that analyzed features such as object speed, direction, and distances between objects.

Researchers Tay et al. [1] highlight the significant role of a widely used model called the Conv-LSTM model to detect anomaly within crowded environments. This is a sort of lightweight model that integrates spatial feature extraction via CNN and temporal sequence learning via LSTM to classify violent and non-violent activities in surveillance videos. In [33], it has shown how the extension of LSTM deals with detecting anomalies. It extracts features from video footage using IoT devices equipped with artificial intelligence (AI). This stage happens in the cloud server. Completing the whole process, it finally goes to the stakeholders who are involved in taking necessary actions. Video is a sequential phenomenon. And the proposed method of [33] mainly detects sequences from it. To do so, this model not only relies on spatial data of an image but also reconstructs that and then classify so that a good result is obtained. This model achieved a maximum accuracy of 95.16% with proper validation. Though it sounds good but has some major issues. Since it's a lightweight model, it cannot deal with complex classification. Moreover, if it needs to work with big data, it may fail as big data requires heavy architecture to bear the pressure of a huge load. And here comes another research done by Esan et al. [12]. It also worked on the same model but didn't use the lightweight extension like [33]. The working method is done by Python with a bunch of necessary libraries such as Tensorflow, Numpy, Scikit-image, Scipy, Matplotlib, Keras, and so on. These tools unlock all the potentiality of this model that helps to handle big data without any trouble. Besides, [33] only determined the accuracy rate of that model but unlike, [12] gave the F1 and AUC scores which made the research more credible. Here the model achieved a score of 89% along with the AUC score of 0.891 and F1 score of 0.94. But here is also an issue that comes from both [33] and [12] that researchers made the architecture from scratch so these are not as advanced as a pre-trained model. However, researchers didn't show how the corresponding manpower who are responsible for taking action will be notified.

Unlike the previous two papers, researchers Sharma et al. [34] came up with a broader solution with their advanced technologies. They not only proposed the CNN-LSTM model but also brought a well-defined system architecture along with this existing model. In the context of the result, [34] got the most accurate result than [33] and [12]. It achieved an accuracy of 98.87% which is undoubtedly a good result. Researchers also introduced another work that is about to notify those people who are connected to take action against people who cause threatening behavior. They made a mobile application using cloud services that are highly scalable. Mainly it allows stakeholders to continuously monitor and keep track of crowd activity. This mobile application has also been successfully developed and is able to provide all kinds of necessary services as every functionality was tested and validated perfectly. However, though [34] is quite better in terms of detection there is a problem with the dataset. Because it is managed by taking data from movies only. So, the lack of data may cause problems in the future. Besides, the paper didn't express how the mobile application will be maintained.

Over the past few years, working on threat detection in the video surveillance system has had huge advancements across many different methods from the videos and LSTM for temporal analysis of those videos, in which the performance is increased by using attention mechanism to focus on significant parts of the video. By following this method they have proved the models effectiveness in recognising interactions among the people. However, their model has limitations, the accuracy of the model reduces for the complex datasets where it is confused by some actions of visual similarities. Moving on to the another model, Thaneshwara and Singh [34] followed a more comprehensive approach, where they focused on crowd analysis and detect abnormal behavior by using open CV for motion analysis which is integrated with CNN and LSTM. This model shows its effectiveness in large scale surveillance systems where the previous model had some limitations. Even though of its effectiveness, the model does not work in highly crowded places because the model has some limitations in recognising small object sizes. In response to the limitations of the model that was proposed, the work of Nasaruddin et al.[11] presented a solution to this problem by introducing Background Subtraction along with 3D CNN model. Their hybrid model addresses the issues that the previous model was facing and they added Visual Attention algorithms to increase the efficiency of the model. Tay et al.[1] paper was limited to smaller datasets, this study can work in diverse environments which makes the paper a more acceptable solution for the real world applications.

Mahdyar Ravanbakhsh et al. [14], in their work, offers a deep learning generative model for crowd behavior analysis using a double conditional GAN. The key concept behind their model is that it is trained on normal data only so that it can generate only what is normal. Exploiting this very insight makes the model handicapped of reconstructing information on unseen or abnormal regions of the data frame. From raw-pixel data optical flow images are generated and vice versa. In this way, the generator learns the motion information and pattern of the input data frame and the author leverages this by comparing the reconstruction with the original data to detect potential anomalies through the discriminator. The authors use AlexNet to record semantic features of the input frames through computing the difference

in real and generated frame’s appearance. A similar approach is also implemented in case of optical flow images to configure the differences. The normalized version of the two differences are then merged together, which provides a better anomaly localization. The researchers also mention the shortcomings of the model, which is, its inability to capture small or obscure abnormal objects. The key strength of their work lies in the elimination of the need for a larger dataset containing abnormal events, which in particular, is difficult to obtain. As for the scope of improvement, the authors indicate the use of Dynamic Images to improve their motion representation. In contrast to the HOF, the author uses MIIs as a feature descriptor, a novelty. In essence, the product of the angle difference between the optical flow vectors and their magnitude between consecutive frames, while compensating for noise and light changes, generates the MIIs. After that, the MIIs is used as input in simple CNN for classification. Unlike in GAN, the latter model is purely discriminative, since it classifies the MIIs directly into normal or abnormal categories. However, the generative capabilities of GAN allows it to have more control over intricate abnormal patterns along with its advanced technique that makes it befitting for real-time surveillance. Both papers’ center of attention is on identifying anomalies on a crowd-level. [6][8]

The previous related works [6][8][13] all focused on local variations, meaning, only using parts of data that are in close proximity for capturing intricate patterns of the dataset. The works of the authors that we are going to review now, introduces Transformer-based architecture, which has the leverage of extracting features across different time steps on a wide range of inputs. It is because the CNN model has a limited receptive field. Since all the aforementioned research integrated the CNN as either the primary method or as the encoder in GAN’s generator part, its limitation is withheld by exploiting the capabilities of the transformer model. C. Yang et al., in their thesis paper named A Transformer-based GAN. Their approach to the problem was a reconstruction based one, where, although satisfactory performance was achieved, couldn’t account for real-time application. While both publications use similar architectural patterns, the latter resolves few issues left by the former one. The authors used the prediction-based approach for anomaly detection. In contrast to vanilla GAN, the researchers incorporated LS-GAN which was less prone to error and more stable during training while generating high-quality images. Despite such a thorough process, the framework is not capable of handling occlusions or obscured objects which can be abnormal. The computational overhead is also a challenge if enough computing resources are not available which is essential and also the dataset has to be extensive as well. [7][4]

As a recent work to mention, a paper was proposed by Ilyas and Bawany [9] proposes a model that consists of the CRAB-NET framework which is integrated with CNN and RNN to analyze crowd behavior in specific scenarios. They use ConvLSTM and LRCN models for the classification of the dataset. The paper focuses on specific crowd scenarios, mainly the urban areas. However, the dependency on small and specific datasets ensures the model’s high performance but the accuracy reduces significantly in diverse environments. Compared to traditional methods, which do feature extraction and classification separately, the approach introduced by [27] is end-to-end that make it more efficient. The model processes sequences of optical flow magnitude images, generated using Farneback’s method to emphasize

motion patterns, employing a dual-branch design that jointly optimizes unsupervised reconstruction (via autoencoder) and supervised classification tasks through a hybrid loss function (weighted MSE and cross-entropy). This approach leverages convolutional layers for spatial feature extraction, LSTM layers for temporal modeling, and global average pooling (GAP) to reduce parameters, achieving 96.07% accuracy on the PETS2009 dataset, outperforming prior methods like dense trajectory analysis (93%). Key innovations include the fusion of ConvLSTM with AE for spatiotemporal representation learning, optical flow inputs to prioritize motion over static appearance, and a single-input-two-output framework enabling simultaneous training. The architecture integrates Convolutional LSTM Autoencoder with optical flow inputs, preprocessing steps, and a joint training strategy combining reconstruction and classification. Key innovations include merging spatial temporal feature extraction and classification into an end to end model. The computational cost of optical, biasness of potential datasets, and grayscale input ignoring colors cues are the few drawbacks of this model. The hybrid structure of the model consists of hybrid training and unified architecture, addressing traditional shortcomings in robustness and computational efficiency.

While CNN-LSTM architectures have demonstrated strong capabilities in capturing spatial-temporal patterns for anomaly detection, their effectiveness is often constrained high computational requirements. These limitations have led researchers to explore alternative techniques such as Zero-Shot Learning (ZSL), which aim to improve model generalization and efficiency. ZSL addresses the challenge of detecting unseen anomalies without requiring labeled samples. A framework called ALFA introduced by the researchers Zhu et al. [17] that critical limitations of existing LVLN-based methods where one is reliance on static anomaly prompts, which suffer from cross-semantic ambiguity, and the other is insufficient pixel-level alignment for accurate localization. They used Run-time Prompts (RTP) to generate context-aware prompts and filter them via a contextual scoring mechanism to mitigate semantic ambiguity and combined global anomaly scores with multi-scale local anomaly maps for final detection. It got an accuracy of 93.2% AUROC (image-level) and 90.6% pAUROC (pixel-level) which is quite good but it has computational overhead as prompt generation and multi-scale processing increase inference time. Actually, models that have complex computation face the same issue. Besides, it tested only on industrial benchmarks; unvalidated on complex real-world scenarios (e.g., medical imaging). Another research [19] proposed a zero-shot anomaly detection framework that synthesizes pseudo-outliers from inlier data distributions to train a binary classifier. This eliminates the need for real anomaly samples during training. The method combines hierarchical feature distillation (via autoencoders and a variational autoencoder) with outlier synthesis from boundary distributions of inliers. The main advantage of this method is robust feature fusion as hierarchical distillation can capture both low-level and probabilistic features. Boundary selection methods of this model provide transparency in outlier generation. Besides, it is compatible with various classifiers (SVM, MLP, RF) and feature sets. In terms of results, this model achieved a 0.976 score, and fusion via VAE improved AUC by 15-20% compared to raw feature concatenation. However, the performance of this model drops on small datasets due to insufficient boundary samples.

Jeongheon et al. [19] proposed a method called WinCLIP for anomaly classification and segmentation in zero-shot or few-shot settings. They have mentioned that traditional methods require training custom models for each task, which is not scalable. WinCLIP uses a vision-language model named CLIP as a base model for zero-shot image-text alignment. The model combines state words and task specific templates to define normal or abnormal behaviour. Moreover it aggregates multi-scale features for segmentation and extracts dense features via sliding windows. An extension of the model, WinCLIP+ stores and retrieves features from normal reference images to compute anomaly scores. As Zero-shot and Few-shot doesn't require task specific training data, that enables scalability across industrial tasks. The model achieves 91.8% and 95.2% image-level AUROC in MVTec-AD and VisA respectively in few-shot settings. Window based feature extraction reduces the computational overhead compared to naive tiling. The model replaces manual feature engineering with compositional prompts which leverage CLIP's semantic alignment. While traditional one-class methods rely fully on normality modeling, WinCLIP+ combines language and visual reference data for computation. Depending mostly on CLIP, performance changes over the model's pre-trained capabilities and prompt engineering quality. Multi scale window processing increases inference time compared to patch-token methods. As the model is not tested on video or dynamic environments, it is limited to static images and requires task specific tuning for the optimal performance. Unlike the traditional methods, PromptAD works for unseen classes by implementing text based descriptions, and can detect anomalies in new categories without additional training. Outperforms traditional art of the state models like DevNet, MVTecAD, WinCLIP, etc and achieves significantly higher AUC scores. When the text descriptions do not perfectly align with the visual anomalies present in the real world data, the CLIP model struggles. PromptAD introduces a novel and effective zero-shot anomaly detection approach by integrating text prompts into the anomaly detection pipeline [21].

Besides traditional zero-shot learning, there is a special type of it called generalized zero-shot learning (GZSL) which is an extension of ZSL. Normal ZSL tests unseen classes based on the seen classes only whereas the generalized version includes only. CBAM enhances feature localization and discriminative power [15]. Also, they used a similarity map generator that aligns local visual features with semantic attributes for regression. These able to bring a harmonic mean of 65.3% of GZSL. However, there may be computational complexity as multiple models (CBAM, Teacher-student networks) have been used here. Researchers Zhou et al. [22] demonstrated AnomalyCLIP, a method to adapt the CLIP vision-language, enabling the model to detect anomalies across diverse domains without requiring target-domain training data. They utilize CLIP's pre-trained text and visual encoders but freeze their parameters and replace object-specific class names with generic terms like "object" and "damaged object" to focus on anomaly semantics. The result of this model evaluated into two datasets. For industrial dataset, it got 91.1% accuracy. For medical datasets, it hits up to 93.4%. Besides, object-agnostic prompts improve performance by 3–15% over object-aware variants. However, the performance of this model drops when the auxiliary data differs significantly from the target domain. Another paper proposed by Joo et al. [25], got 95.9% pixel-level AUC on VisA dataset, and 92.3% pixel-level AUC on MVTec dataset. No prior data is required from target categories

for their Zero-shot generalization, and it has adaptive learning capabilities instead of manual engineering. Even though they have achieved great milestones, the quality of fine-grained descriptions depends on LLM capabilities, and multi-scale processing in MMCI increases inference time.

Another line of research explored future frame prediction instead of reconstruction. Rather than autoencoding the same frame, these methods predict what the next frame should look like given the past frames, using architectures like convolutional LSTMs [27] or predictive autoencoders. If the predicted frame diverges from the actual next frame, the assumption is that an unforeseen event (anomaly) must have happened. Researchers presented multiple instance learning approach where videos are labeled at the video level as normal or anomalous, but not where the anomaly occurs. Their method uses a ranking loss to ensure anomalous videos have higher anomaly scores than normal ones on average. Later, they optimized an autoencoder for reconstruction and a future frame prediction task. By multitasking, the model learned more robust representations, achieving better generalization. Such approaches hint at the benefit of learning multiple aspects of normal behavior simultaneously.

However, recent studies have begun to address context-awareness. Sun and Gong (2023) [30] proposed a scene-aware anomaly detection technique using a hierarchical semantic contrast (HSC) method. Another notable work is by Yang and Radke (2024) [32], who introduced a context-aware framework called Trinity for long term video anomaly detection. They explicitly model three modalities, appearance, motion, and context. In their formulation, context includes information such as time-of-day or event schedules. The integration of semantic information through vision-language models is a very recent and promising development. CLIP [29] introduced a powerful paradigm by training on millions of image-caption pairs to create a joint image-text embedding space. These models achieved good benchmarks in terms of AUC, indicating that the rich semantic prior of CLIP helps distinguish complex events better than purely data-driven features.

This thesis focuses on the success of contrastive learning and predictive modeling as seen in CPC and TimeSformer literature to create a contextually informed, zero-shot anomaly detection model. In doing so, it contributes to the literature by integrating these strands and pushing the performance and applicability of VAD systems closer to real world requirements.

Chapter 3

Proposed Methodology

3.1 Methodology Overview

Our proposed method is a unique approach for detecting anomalies in surveillance videos without needing any examples of abnormal behavior in the training segment. The main idea is to make the model both context-aware and capable of understanding how events normally occur over time. By using contrastive learning and predictive modeling as complementary techniques we are able to achieve this. These are implemented as dual-stream deep learning architecture, where one stream processes the video input and the other handles context information like time, day or scene descriptions.

The video stream uses a [6][7]transformer based model called TimeSformer, which extracts detailed spatiotemporal features from the video [3]. This helps the model understand what is happening in the video across the time. On the other hand, the context stream turns auxiliary data into meaningful embeddings. These might come from a text encoder like CLIP which can understand semantic context that processes metadata. The outputs from both streams are then combined into a single representation that reflects what is happening in the video which is adjusted for the specific context. This fusion allows the system to determine whether a behavior is appropriate for a given situation or not. To train this model, we used two types of loss functions. The first one is contrastive loss, which teaches the model to match video and context pairs that belong together and separates those that don't match. For example, a video showing people gathering at night in an office building might be flagged as suspicious because the context suggests the office should be empty. The second loss is predictive which is based on the idea of Contrastive Predictive Coding (CPC)[15]. The model tries to predict what will happen next in the video not by guessing pixels but by predicting future features. If the prediction is far off from the normal, it suggests something abnormal occurred like sudden movements or unexpected object appearances.

Our system is trained entirely on normal video segments paired with their proper context where it never sees any abnormal data while training. This is what makes it zero-shot, it learns only what is normal and identifies anything different as abnormal. During application, the model generates two scores. The first is a context alignment score that measures how well the video matches the expected context. The second

is a predictive score that reflects how well the model could anticipate the next part of the video. A low context alignment score or a high prediction error signals an anomaly. These scores can be combined into a final anomaly score to make the decision. This combination of techniques enables the model to identify both context-based anomalies, like people appearing in places they shouldn't be at that time, and temporal anomalies, like unusual actions or motion patterns. The use of a pre-trained vision language model like CLIP further helps the model recognize concepts it has never seen during training by linking visual patterns to high-level ideas. As a result, the system is able to generalize well and flag unexpected events without needing labeled examples of anomalies. This method provides a robust and flexible solution for real-world anomaly detection in surveillance settings.

3.2 Proposed Model Architecture

The proposed Context-Aware Zero-Shot Anomaly Detection framework is a unified model that jointly learns spatiotemporal representations and semantic context for surveillance videos. The architecture consists of four main components, a global scene encoder for capturing holistic video context, a predictive modeling module for anticipating future scene dynamics, a context-conditioning network to modulate predictions based on scene context, and a CLIP-based text encoder to inject high-level semantic knowledge. These components are trained together in an end-to-end manner to learn a shared embedding space for video and textual context, enabling zero-shot inference. Figure 3.1 illustrates the architecture. Input video frames are processed by the TimeSformer-based encoder, whose outputs feed into the DPC-RNN predictive module. A context-conditioning subnetwork takes global scene features or associated text descriptions to produce modulation parameters that inform the predictive module. In parallel, a text encoding branch using CLIP's language encoder provides a semantic context vector. All feature streams are projected into a common embedding space and optimized with a contrastive InfoNCE loss and a hybrid Contrastive Predictive Coding (CPC) loss [2]. This design allows the model to learn what constitutes "normal" patterns in a scene and detect deviations as anomalies without explicit anomaly examples, in a zero-shot way. Next, we detail each module and the training methodology.

Zero-Shot Learning (ZSL)

What if a machine learning model could learn and work without ever needing labeled data? This is precisely where Zero-Shot Learning comes into play. It is a machine learning algorithm that enables a model to identify or classify new instances of data or concepts that it had not previously seen or was not explicitly trained on such labeled data as well as having the ability handle data for which it wasn't particularly trained. It allows the model to recognize things that it had never encountered. It utilizes the information it already knows and connects it to the new situation, which was presented in front of it. Let me explain this with an example: Suppose a model is trained to recognize animals but not particular ones like Zebra in its training phase. In case of other machine learning approaches, the model will fail to classify an animal being Zebra because it was never explicitly shown any labeled data of Zebra during its training period. But ZSL will be able to figure it out by using

description even though it was also not shown any examples of it. The ZSL model knows about the information of animals from its training phase like 'they have four legs,' 'has stripes' or 'has a tail'. The ZSL model is given a new description that reads as, 'A horse is an animal with black and white stripes and a tail that lives in Africa.' Using its knowledge on animal attributes and characteristics along with the given description it will successfully deduce that the animal is in fact a Zebra even though it had no prior knowledge about it during its training. This is the power of Zero-Shot Learning. It uses these auxiliary information to its advantage.

Contrastive Predictive Coding (CPC)

This is an unsupervised learning approach that extracts useful representations from high-dimensional data. The key idea behind this is to learn such representation by predicting the future latent feature representation through the use of autoregressive models. The key insight here is that using past contexts, CPC predicts future parts of a sequence but not by directly generating the data but by distinguishing the true future from the false ones, a learning procedure known as contrastive learning. In this case, a probabilistic contrastive loss is used alongside negative sampling. It learns rich temporal data without labels, which is ideal for temporal modeling. **Equation 3.3**, $\mathcal{L}_{align+pred}/\mathcal{L}_{total}$, shows how we combined these two losses for our particular task.

TimeSformer for Spatiotemporal Features

To capture the overall appearance and movement patterns in surveillance videos, we use TimeSformer, a transformer-based model designed specifically for video understanding[8]. TimeSformer treats a video not as a single image sequence but as a series of smaller visual patches. Each frame in the video is split into patches, which are then turned into vectors through linear projection[10]. To help the model understand the position and order of these patches, we add spatial and temporal position information to each one.

These processed patches form a long sequence of tokens that are passed through multiple transformer layers. Through self-attention mechanisms, the model learns how different patches relate to each other across both space and time. This helps the model capture long-range dependencies and understand the broader context of the video clip. We use a specific variant of TimeSformer known as "divided space-time attention," which separates the spatial and temporal attention computations to make the model more efficient, especially for longer video clips. The result of this processing is a rich sequence of feature vectors that describe the video at each moment. In our implementation, we use these outputs in two main ways. First, the feature from each frame is passed to a predictive model that learns how the scene evolves over time. Second, these features are mapped into a shared embedding space so that they can be compared directly with semantic context information from text descriptions.

By using TimeSformer, our model can pay attention to all parts of a scene at once. This is especially important in anomaly detection, where unusual events often involve complex interactions or context-sensitive behavior. Unlike traditional convolutional

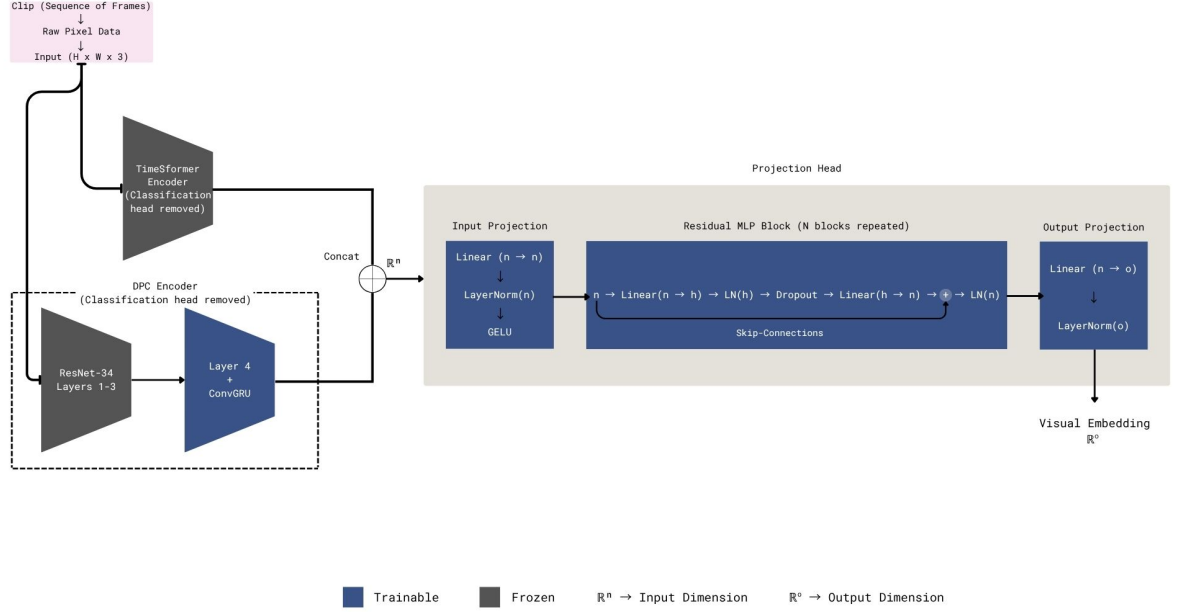


Figure 3.1: Video Architecture Pipeline

networks which focus on local patterns, TimeSformer’s global self-attention enables it to understand the full picture. This makes it ideal for modeling what is “normal” in different surveillance scenarios.

Temporal Predictive Modeling with DPC-RNN

While the TimeSformer encoder extracts features from each frame in the video, it does not capture how the scene evolves over time. To model this temporal progression, we used a Dense Predictive Coding Recurrent Neural Network (DPC-RNN) [24]. This module takes the sequence of frame-level features produced by the TimeSformer and learns how to predict what should come next in a normal sequence of events. At every moment in time, the RNN updates its internal memory using the current feature and the previous hidden state. This updated state acts as a summary of everything the model has seen so far. The predictive head then uses this state to forecast future representations of the video. Instead of trying to generate actual video frames the model predicts the high-level features that represent the expected future. The goal is to match these predicted features as closely as possible with the actual ones that occur later in the sequence.

Figure 3.1 illustrates how the video pipeline is structured in its entirety. Layer4 & ConvGRU of DPC-RNN is fine-tuned for domain adaption. Two separate inputs enter both the feature extractor backbones, varying in terms of the number of frames that the backbones anticipate. Their initial output embeddings are concatenated and is then passed to the residual MLP projection block, where it will be projected to the CLIP’s space. We used multiple residual blocks in the projection head, which will help make the training stable and the transmission of information and gradients easier. Furthermore, the Projection Head will learn to project its output into CLIP’s space in such a way that visual embeddings will land closer to its correspond-

ing textual embedding and further away from negatives. Firstly, inside the input projection, the concatenated visual embedding passes through a Linear layer, which is a FCNN or a Dense layer, and the layer will learn a Linear transformation of the input features. The output from this is then normalized using Layer Normalization, which helps to stabilize the training process. Also ensures that the network’s layer receives data information with a consistent distribution with a mean of 0 and standard deviation of 1. A Gaussian Error Linear Unit (GELU) is used to introduce non-linearity into the model. The Residual MLP Block is the core processing unit of the Projection Head. As mentioned above, N residual blocks are used, which helps the model to learn progressively more abstract and refined features of the input as a result of stacking of these blocks. The arrow from the input to the final output of this block that bypasses the inner layers, LN, activations is a ‘skip’ or ‘residual’ connection. This is the essence of a Residual Network (ResNet) which solves the vanishing gradient problem. If we trace the path inside this block, the output from the Input Projection block is passed through a Linear layer which projects it to a higher dimension h allowing the model to learn more richer representation. This a h -dimensional hidden vector. LN is applied to it. Also a regularization method is used here, allowing the model to generalize better to unseen, new data. After that the h -dimensional hidden vector is projected back to the original dimension. A skip-connection allows for the original input to be added to the output of this final linear layer. The entire sequence of this block is repeated N times. It is then normalized and passed to the Output Projection block as input. The input is passed to through another linear layer where its projected to the specific dimension and normalized and thus produces the visual embedding, where $\mathbf{v} \in R^d$.

The training strategy follows a contrastive approach which is using the infoNCE loss. The model learns to make its predicted future feature similar to the actual future feature while making it dissimilar from unrelated ones. These unrelated or negative examples are typically taken from other parts of the same video or from different videos in the same batch. This way, the model learns meaningful patterns that consistently appear in normal videos, while ignoring small random changes like noise or slight movement that don’t indicate anomalies. By summing the predictive loss over multiple future steps and across time, the DPC-RNN becomes skilled at learning how scenes usually unfold [25]. This is useful for detecting anomalies because during the training phase the model only sees normal video clips. If something unusual happens later it won’t be able to predict it accurately, resulting in a large difference between the predicted and actual features. This prediction error becomes a clear signal that something unexpected is happening in the video.

In this way, the DPC-RNN serves as a temporal normality checker. Its ability to correctly anticipate future events during normal operation helps the overall system identify when something deviates from the expected pattern. During inference, we use this prediction error to help flag anomalies.

Context-Conditioning Network for Scene-Aware Modulation

A key innovation in our model is its ability to adjust its behavior based on the specific scene it is observing. In real world surveillance, what is considered “normal” varies greatly from one location to another [32]. For example, heavy pedestrian ac-

tivity might be typical in a shopping mall but highly unusual in a restricted area. To account for this, we use a Context-Conditioning Network that adjusts the model’s predictions according to scene-specific information.

The context used to guide the model can come from two sources, visual features and textual descriptions. Visually, the TimeSformer encoder captures static aspects of a scene, such as its layout or common background elements. Textually, we can provide a natural language description of the scene which is encoded using the CLIP text encoder. These inputs are processed to form a context vector that is used to adjust how the prediction model behaves. In our case, raw pixel data from a single mid-frame of a clip is provided to the Context-Conditioning Network as its input. It outputs a context vector embedding and is passed through a ContextGate Block which is FCNN or MLP. We introduced a learnable parameter here, β , as illustrate in **Figure 3.2**, which is used to control how much information will be passed to the textual embedding so that not all its own information is washed out by the context vector. The ContextGate learns over time as to how much information is needed by the textual embedding to understand the surrounding context or whether it is capable of determining the context by itself or not. It is then fused with the text embedding through residual addition. Alternatively, the context vector can be directly added to the visual features before transferring it to the RNN. These adjustments help the model make scene-specific predictions and reduce false alarms.

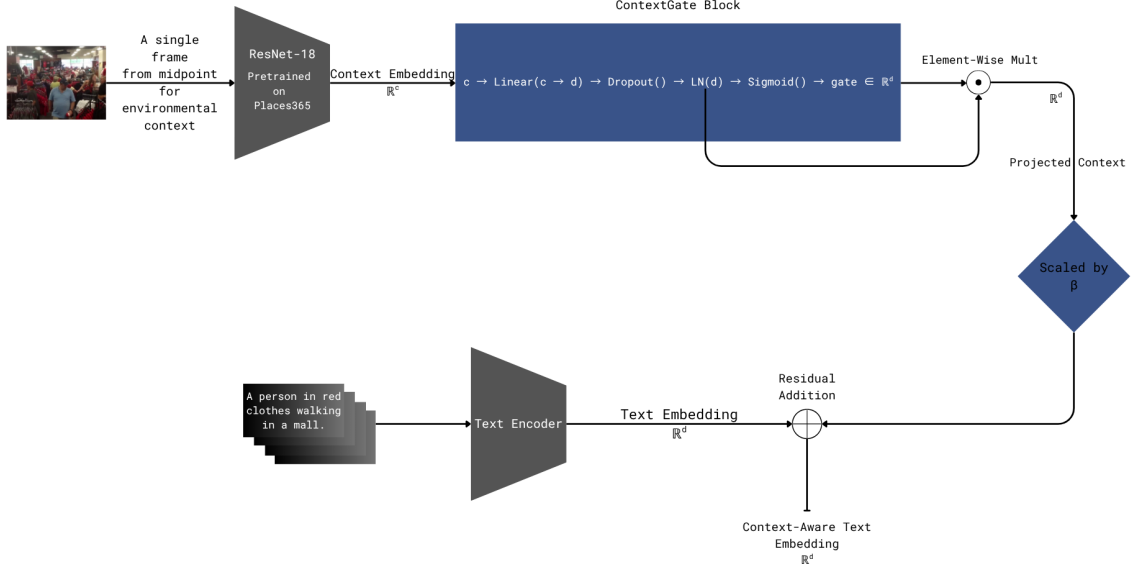


Figure 3.2: Text Pipeline

This design enables the model to adapt to new environments without retraining, which is essential for zero-shot anomaly detection. It allows the system to understand what kind of behavior is expected in a given context and to detect deviations effectively.

CLIP-Based Text Encoder for Semantic Context

To enhance the model’s ability to understand high-level semantics and support zero-shot anomaly detection, we incorporate a CLIP-based text encoder. CLIP is a powerful model trained on a vast amount of image-text pairs. It can map both images and natural language descriptions into the same feature space, which allows for a meaningful comparison between visual and textual content.

In our system, we use the CLIP text encoder as shown in **Figure 3.2** is used to convert descriptive sentences about a scene into feature vectors. These descriptions define what is typically expected in a given setting. Each sentence is transformed into a semantic embedding that represents the normal behavior or appearance of a specific scene. Rather than relying on CLIP’s image encoder, we train our TimeSformer-based video encoder to produce embeddings that align with these text representations. During training, we use contrastive learning to bring the video embedding closer to the matching context text and push it away from unrelated ones. This teaches the model to understand and align the video content with the semantic expectations.

This text-guided alignment gives our system strong zero-shot capabilities. Even if the model never saw a specific anomaly during training, it can still recognize it by detecting a mismatch between the video and the expected context description. The CLIP based text encoder acts as a form of semantic supervision that guides the model to understand what is normal and helps it to detect when something deviates in a meaningful way.

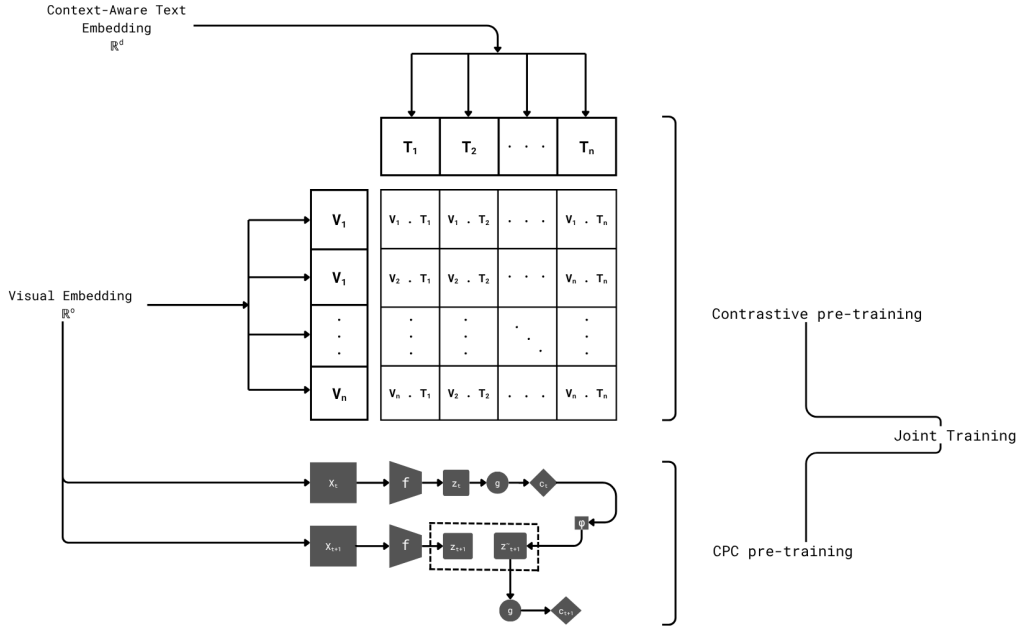


Figure 3.3: Joint Training on Prediction Loss & infoNCE

Joint Feature Fusion and Training Objective

Among the major components of our system, the TimeSformer feature extractor backbone is kept frozen, only the last two layers of DPC-RNN predictor are fine-tuned for domain adaption and the rest are kept frozen, the context-conditioning module is learnable, and the CLIP text encoder is kept frozen, the residual MLP projection is trainable and together using a unified training loss we effectively train our anomaly detection system. Feature fusion happens at several points in this architecture to ensure that the model can leverage both visual and contextual information.

Firstly, the input raw pixel data from varying frames are sent to both TimeSformer and DPC and later their feature output is then concatenated and projected into the CLIP’s embedding space where the textual embeddings already resides. The projection is done using residual MLP which ensures that the model can capture complex patterns. Also, the residual addition provide the MLP a shortcut which is that if the layer by layer processing is not useful the FCNN can retain its original input without little to no changes through the use of skip-connections as illustrated in Projection Head section in the **Figure 3.1** while in other cases it adds a small change instead of complex transformation at every layer. This small tweaks to the input makes the learning faster and easier. This determines whether a layer is useful or not and whether it will learn useful changes or not, acting like a memory lane for gradients and information, helping them flow easily.

Later, we project the outputs of both the video and text branches into a shared embedding space, ensuring they are directly comparable. We use learnable linear layers to adjust the video features so they align with the fixed dimension of CLIP’s text embeddings. The predictive loss teaches the RNN to forecast future visual representations of normal events, while the contrastive loss ensures that the video features align with the correct text description which is depicted in the **Figure 3.3** where the selected layers and models are trained on both training objectives and the weighted sum of the two backpropagated infoNCE-style objectives. **Equation 3.1** $\mathcal{L}_{\text{align}}$ is the classic contrastive term between visual embedding and correct textual embedding.

$$\mathcal{L}_{\text{align}}(i) = -\log \frac{\exp(\text{sim}(v_i, t_i)/\tau)}{\sum_k \exp(\text{sim}(v_i, t_k)/\tau)}. \quad (3.1)$$

Here, with each sample i , consisting of the visual embedding v_i and the positive textual embedding t_i along with a set of negatives t_k . The training objective here is to bring the positive visual-textual pair closer to each other while pushing away the negatives as much as possible.

The Contrastive Predictive Coding (CPC) on the other hand, also implements the core infoNCE loss but instead of video-text pairs, they predict future latent representation z_{i+1} from a given context embedding c_i at time-step i and

$$\mathcal{L}_{\text{pred}}(i) = -\log \frac{\exp \left(f(c_i)^\top z_{i+1} \right)}{\sum_k \exp \left(f(c_i)^\top z_k \right)}, \quad (3.2)$$

pulls positive pairs, that being, the prediction and the ground truth at next-step $i + 1$ while simultaneously pushing the negative set z_k further away. **Equation 3.2** $\mathcal{L}_{\text{pred}}$ illustrates the prediction loss. $f(\cdot)$ is the small MLP used here for prediction. Both losses are combined into a single-joint loss which is scaled by α , deciding how much weight each loss will contribute during training and testing.

$$\mathcal{L}_{\text{total}} = \alpha \mathcal{L}_{\text{align}} + (1 - \alpha) \mathcal{L}_{\text{pred}} \quad (3.3)$$

The typical value for the weighting factor as illustrated in **Equation 3.3** α is 0.5.

The CLIP text encoder remains fixed to preserve its pre-learned semantic knowledge, while the rest of the model is fine tuned. By the end of training, the model has learned to group normal behavior patterns properly around their corresponding context descriptions which allows it to flag any deviation as an anomaly without ever having seen abnormal events during training while also being able to use the prediction loss for a surprise during the testing flagging behaviors early that strays away from the normal line of projection.

Adaptive Inference and Scene-Conditioned Anomaly Detection

During inference, our model processes video frames continuously to compute an anomaly score at each time step. This process is adaptive in two key ways, it can analyze longer video sequences than it saw during training, and it adjusts its behavior based on the specific scene being observed. While training is done on short clips due to memory limitations, the DPC-RNN used in our model is capable of maintaining context over longer sequences. During the test time, we allow it to accumulate temporal information from extended video streams that improve prediction stability and reduce false positives.

Because the model is context-aware, we use scene specific information during inference. For each camera or scene, we input the corresponding text description so that the model understands what kind of behavior is expected. The model compares what it sees with what it expects in two ways, firstly it checks how different the actual frame is from what it predicted, and secondly it checks how well the current scene matches the provided context description. If either of these checks indicates a strong mismatch, the model flags that moment as anomalous.

Anomaly scores are calculated by combining the prediction error with the deviation from the expected context. Thresholds for detection can be tailored to each scene using normal video segments to reduce false alarms. Furthermore, because we use CLIP’s shared embedding space, the model can not only detect anomalies but also potentially explain them by comparing the video to various text descriptions. This capability adds depth to the zero-shot detection framework and allows for more informed interpretation.

3.3 Dataset Description

In real life, threat incidents don't occur frequently which brings up a critical challenge in front of the researchers when it comes to collecting data that consists of abnormal behavior. To prevent this issue, we take an alternative approach where we will work with the datasets of normal behavior during the training phase and then apply that knowledge to detect anomalies. There are plenty of existing datasets related to normal behavior that are freely available in Kaggle such as the one published by H. Sanskar; UCF Crime Dataset. This dataset consists of extracted frames from full-length videos aimed at real-world anomaly detection in the surveillance system. While the UCF Crime Dataset also includes the classes of anomalies, we will specifically select only the class representing normal behavior. In terms of size, the training subset consists of 589 untrimmed videos, while the testing subset comprises 107 untrimmed videos. The larger our dataset becomes, the better our model will learn, which ultimately helps improve its performance during the testing phase. Instead of using frames, we use raw videos directly. To be specific, we used the UCA dataset which is an extension of the original UCF Crime Dataset with the addition of textual description. These descriptions are paired with specific portions of each video, almost like a clip-text pair. we will utilize this as the base for our Zero-Shot Learning by using Contrastive Loss to teach our model semantic alignment with normal behaviors.

3.3.1 Data Handling

We meticulously handled the data based on our comprehensive and systematic methodology, ensuring that every aspect of the analysis was conducted with precision and attention to detail. Given the continuous and high-volume nature of surveillance video streams, storing every frame individually would be prohibitively resource-intensive and impractical for large scale experiments. To address this, we adopt a structured and manifest-driven approach that enables dynamic clip extraction from compressed video files during both training and testing phases.

The data handling process begins with a well-organized manifest file that serves as the foundation for data annotation and retrieval. Each entry in the manifest consists of four key components, including label, video ID, start and end timestamps, and event description. Here label indicates whether the clip represents a normal or anomalous activity and video identifier uniquely references the source file. Also, start and end timestamps help to denote the temporal boundaries of interest and event description used to describe the observed scene. This compact structure allows the pipeline to target specific segments within longer video streams without the need for exhaustive manual segmentation or storage.

To avoid storing each video on RAM, which would require extensive storage and slow down processing, we utilize a lightweight and high-performance video decoding library called Decord. Unlike traditional preprocessing pipelines that rely on storing pre-extracted frames, leading to high memory usage and slow loading times, Decord decodes video frames directly from the compressed file in real time. This on-the-fly decoding is made possible by converting the annotated timestamps into frame

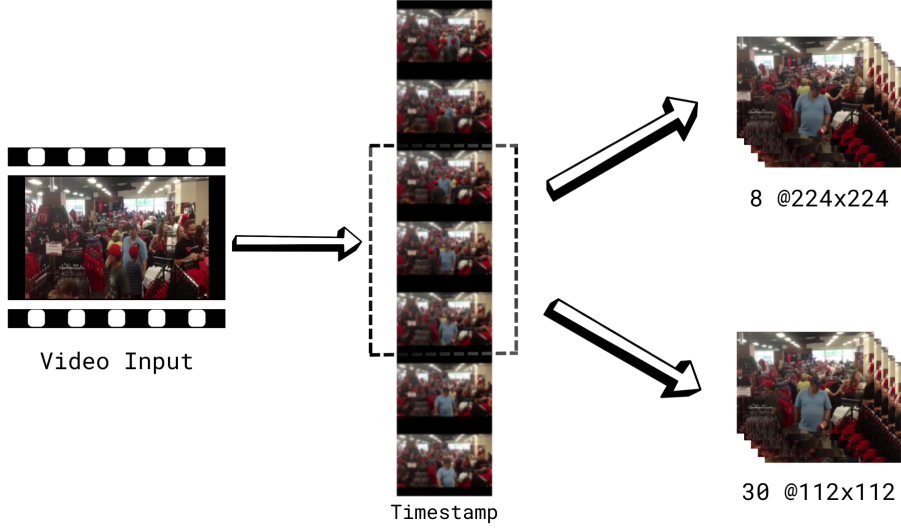


Figure 3.4: Decoding Raw Pixel Data

indices based on each video’s frame rate, allowing precise extraction of relevant segments without unnecessary overhead. In the **Figure 3.4**, for each annotated event, we sample two sets of frames - one is a short high-resolution clip (**8 frames at 224×224 pixels**) used by our TimeSformer model to capture scene-level context, and another is a longer low-resolution clip (**30 frames at 112×112 pixels**) used by our DPC for predictive temporal modeling. We did it using uniform sparse bin sampling to ensure coverage and mitigate temporal redundancy, along with jitter that ensures frames are picked at random from each bin, which in turn, is also useful since Timesformer trained on sparse sampling. After that, the decoded raw pixel data are then passed to the timesformer and DPC.

3.3.2 Exploratory Data Analysis

To ensure temporal consistency and enable efficient processing, the dataset undergoes a systematic frame extraction process prior to analysis. This preprocessing step is critical for maintaining uniformity across all video samples, particularly in tasks involving temporal feature learning. By standardizing the frame selection, we eliminate inconsistencies that may arise due to varying frame rates or irregular sampling patterns inherent in raw video data. The dataset has a uniform sampling rate for every video. The frame gap is consistent across all the videos. It means the video frames are captured at regular intervals. Each frame is followed by the previous one by exactly 10 frames, meaning that every 10th frame is extracted only. This reflects that the dataset is processed in a structured manner to ensure consistency for computational efficiency. From the heatmap below, it is evident that the number of frames skipped between frames is strictly kept to 10 to ensure there is no redundancy in the frame distribution and irregular frame timings.

The difference between each consecutive frame effectively tells us about the dynamic motion happening over time in each video. The purpose of this analysis is to showcase how much each video changes from frame to frame as time passes. The SAPD graph effectively captures the temporal changes. From the graph below, we can see

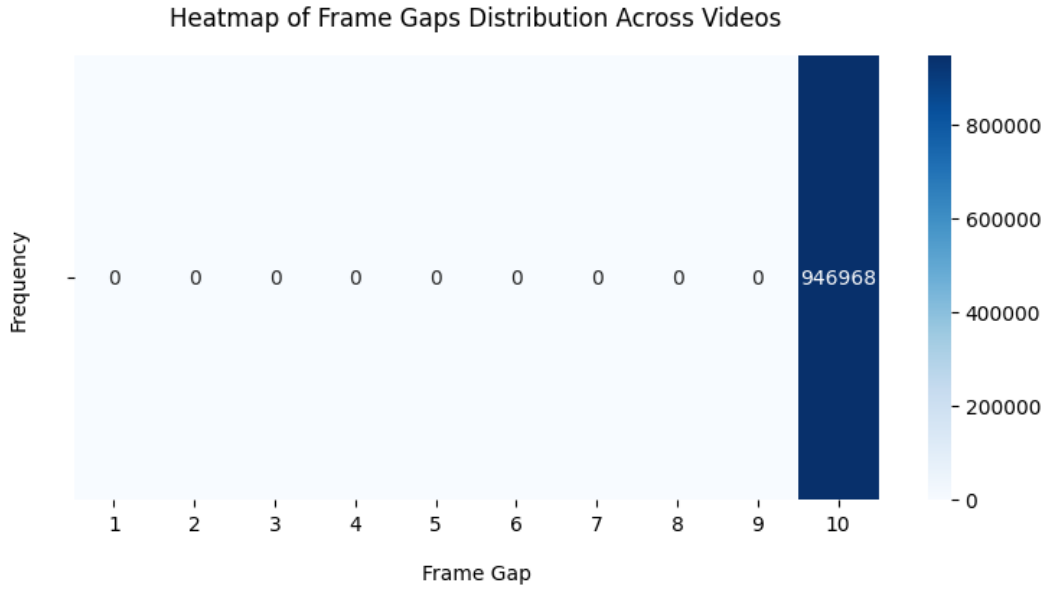


Figure 3.5: Temporal Frame Gaps Analysis

how the pixel differences evolve across videos. It focuses on providing the overall frame-level changes by combining all the pixel-level differences into a single scalar value for each frame. For some videos, the graph stays comparatively flat with somewhat noticeable pixel differences.

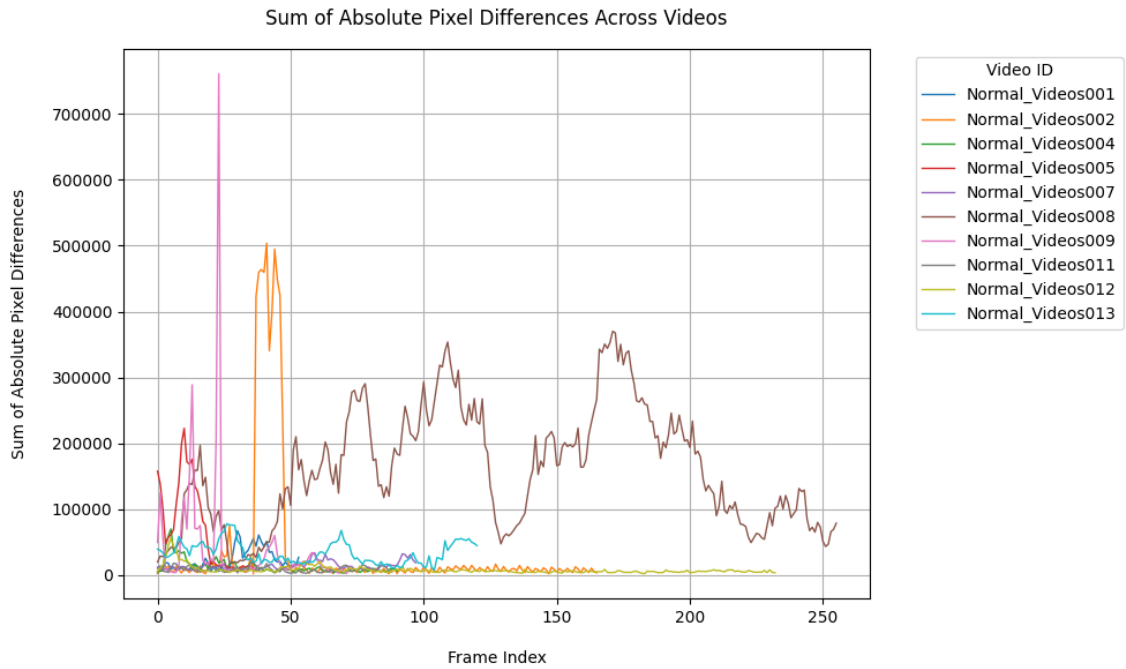


Figure 3.6: Frame-to-Frame Difference

This could indicate nominal motion changes among those sequential frames. In contrast, some videos exhibit sharp spikes, suggesting that there are rapid changes in motion or transitions between scenes in those consecutive frames. Some curves are steep while some decline gradually and some remain flat.

Chapter 4

Result Analysis

4.1 Our Contributions

We have contributed in three key areas of our proposed methodology. The first key contribution is that our entire end-end pipeline is trained purely in a Zero-Shot manner. That is, we have achieved pure Zero-Shot Learning through our own *modus operandi*. We did not expose our model to any other information beyond normal behavior, which is what it was trained entirely on. This is what we call true Zero-Shot Anomaly Detection (ZSAD). We were able to achieve this because of a joint training objective that we implemented for our model. This is the second key contribution of our work, which is adapting the alignment loss and prediction loss from contrastive loss. The former being the core to CLIP’s ZSL approach, and the latter from Contrastive Predictive Coding (CPC). Using this combined training objective is what led us to attain a pure Zero-Shot scenario. Our last contribution lies in adapting the textual embeddings to the surroundings of the events described, effectively making the textual embeddings context-aware and in turn rendering our whole pipeline the same. This key addition makes the model capable of differentiating place, time, and scenarios with respect to the events that occurred and in turn helped reduce false positives/alarms.

4.2 Results and Comparisions

To evaluate the result of our proposed model, we performed a comparative analysis against several methods that have already worked on the UCF-Crime dataset, a benchmark that is widely used for anomaly detection in surveillance scenarios. We focused on key evaluation metrics ROC-AUC and PR-AUC, which are essential for measuring a model’s ability to distinguish anomalies, especially in the context of unbalanced surveillance data.

The **table 4.1** below shows a comparison of different models that trained on the UCF Crime Dataset using ROC-AUC and PR-AUC as evaluation metrics. ROC-AUC metrics measure the model’s ability to distinguish between normal and anomalous events, while PR-AUC is particularly important in this context due to the inherent class imbalance in surveillance data, where normal events significantly outnumber anomalies. Among these models, Flashback (ViT-L) achieved the highest score of 87.3 % ROC-AUC and 75.1 % PR-AUC. But the problem is that it relies solely on

Table 4.1: Comparison of ROC-AUC & PR-AUC score for Anomaly Detection Models with Zero-Shot Learning on UCF-Crime dataset

Model (Approach)	Pure Zero-Shot?	Text?	ROC-AUC $\uparrow$	PR-AUC $\uparrow$
Flashback (ViT-L)	No	No	87.3 %	75.1 %
Our model (CLIP + DPC + TSF)	Yes	Yes	84.5 %	72.3 %
AnomalyCLIP (ViT-B/16 + CLIP)	No	Yes	82.4 %	68.7 %
ViT-I3D	No	No	72.1 %	57.2 %
Inflated 3D-CAE (I3D)	No	No	68.0 %	51.4 %

visual features from a large ViT-L backbone without incorporating semantic textual information. On the other hand, our hybrid model that combines TimeSformer, DPC-RNN, and CLIP achieved a competitive score of 84.5 % ROC-AUC and 72.3 % PR-AUC. Though this model is slightly behind the Flashback model, it stands out as the best among zero-shot and vision-language approaches. Compared to AnomalyCLIP, which gained 82.4 % ROC-AUC and 68.7 % PR-AUC, also uses CLIP but lacks predictive and spatiotemporal modeling, where our model demonstrates clear improvements, attributed to its integration of DPC-RNN for future prediction and TimeSformer for temporal encoding, along with context-aware modulation.

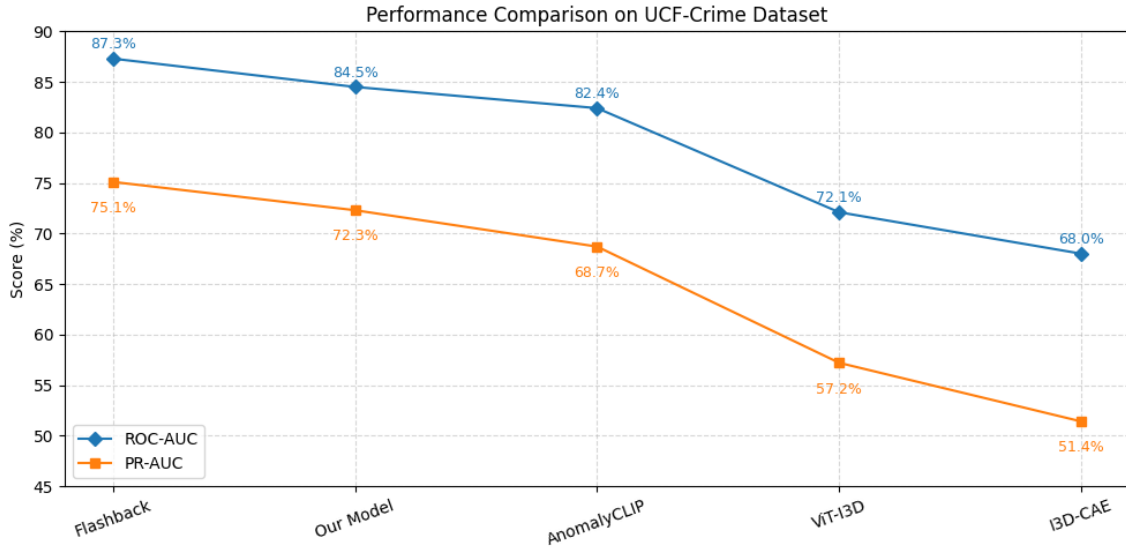


Figure 4.1: Performance Comparison on UCF-Crime Dataset

However, ViT-I3D and Inflated 3D-CAE, which are considered as traditional vision-only models, performed significantly worse compared to our proposed model. So it is clear that our model offers a strong balance of performance and generalization, achieving impressive and outstanding results among zero-shot approaches while also maintaining high anomaly detection precision.

Table 4.2: Comparison of mAP(%) & Detection Delay(s) for Anomaly Detection Models (with and without Zero-Shot Learning)

Model (Approach)	ZSAD?	mAP(%)	Detection Delay(s)
Our model (CLIP+DPC+TSF)	Yes	62.5 %	0.45s
Multimodal Asynchronous Hybrid Net	No	54.2 %	0.05s
Rethinking VAD (Continual Learning)	No	48.7 %	0.60s
VADA/RTFM-style (Weak Supervision)	No	52.1 %	0.80s
Flashback (Memory-Driven) ZSAD	Yes	45.5 %	1.2s

In the **table 4.2** above, a comparative evaluation shows that our proposed hybrid model, combining TimeSformer, DPC-RNN, and CLIP, achieves outstanding performance by balancing accuracy and responsiveness. It gains a high mean Average Precision (mAP) of 62.5 %, surpassing notable models like Flashback (45.5 % mAP) and Multimodal Asynchronous Hybrid Net (54.2 % mAP). Additionally, the detection latency of our model is notably low at approximately 0.45 seconds, considerably faster than Flashback (1.2s) and similar approaches such as Rethinking VAD (0.60s) and VADA-style weakly supervised models (0.80s). While Multimodal Asynchronous Hybrid Net achieves exceptional latency (0.05s), as it relies on specialized event-based hardware, limiting its broader applicability.

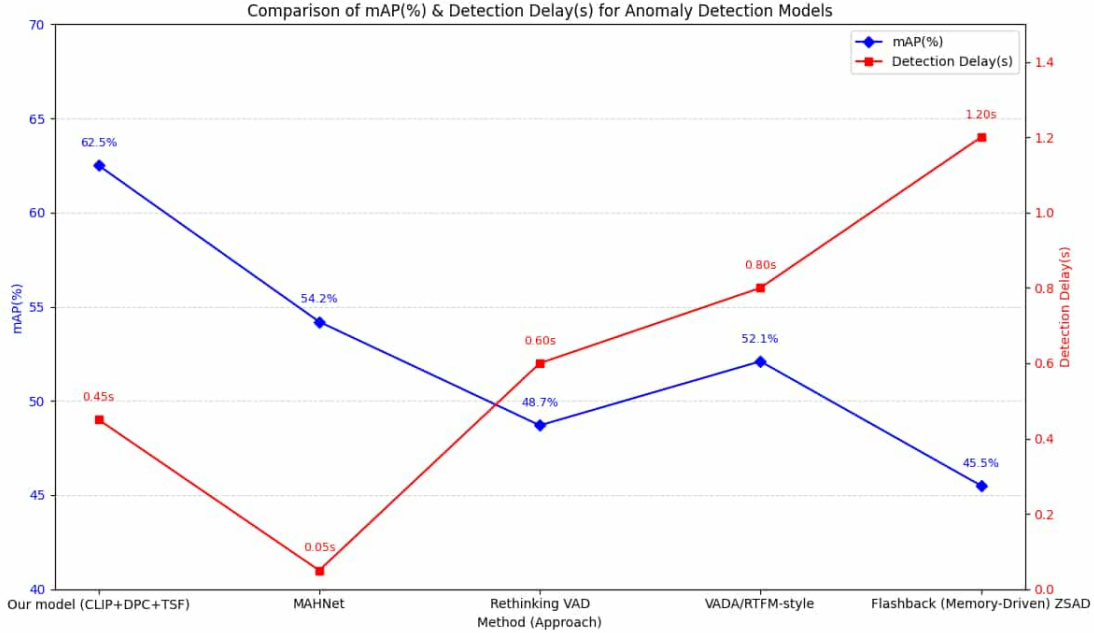


Figure 4.2: Comparison of mAP and Detection Delays for Anomaly Detection Models

On the other hand, our zero-shot model maintains competitive detection speed using conventional hardware. It effectively balances unsupervised anomaly detection ca-

pability and real-time practicality, making it particularly suitable for diverse surveillance and real-time anomaly detection scenarios.

Table 4.3: Comparison of F1-score for Anomaly Detection Models (with and without Zero-Shot Learning)

Model (Approach)	ZSL?	F1-score
Our model (CLIP + DPC + TSF)	Yes	0.74
VADOR (Temporal segmentation, TALNet)	No	0.63
ADNet (Temporal convolutions)	No	0.58

In the **table 4.3** above, we provide a comparative analysis of the anomaly detection performance in terms of the balanced F1-score metric between our proposed zero-shot approach and other non-zero-shot approaches. Our model achieves the highest F1-score, which is 0.74. It significantly outperforms leading methods like VADOR and ADNet, which achieved F1-scores of 0.63 and 0.52, respectively, when evaluated at 25 % and 10 % temporal overlap.

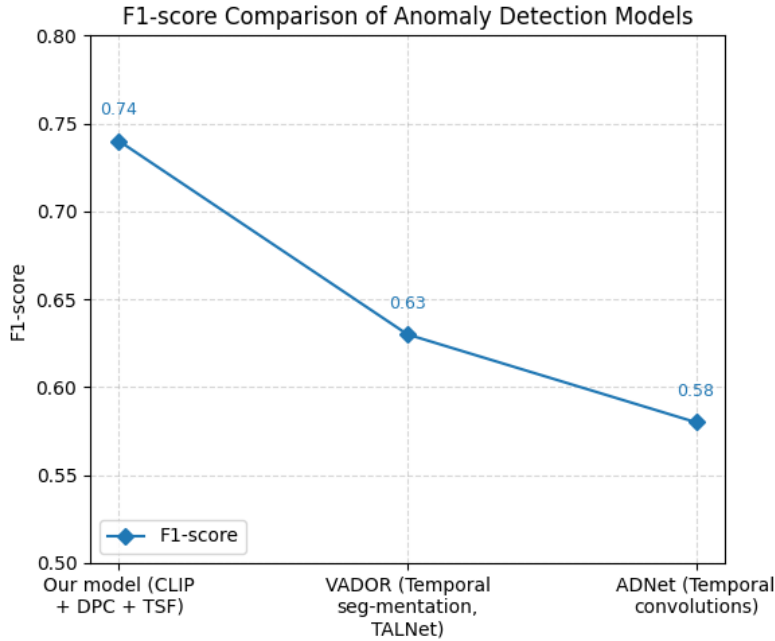


Figure 4.3: F1-score Comparison of Anomaly Detection Models

Notably, unlike VADOR, which relies on temporal action localization and ADNet’s temporal convolutions, our zero-shot approach integrates semantic guidance from CLIP embeddings alongside dense predictive coding and TimeSformer-based spatiotemporal encoding. This combination not only ensures superior detection performance and balanced precision-recall but also offers robust generalization to unseen anomaly types without the need of retraining.

4.3 Ablation Study

Ablation study is done to dissect our entire architecture and evaluate each module separately to understand how much each module contributed to the entire pipeline. This is used to determine whether the modules used provide any useful gains or just an overhead to the overall architecture. The **table 4.4** below shows the progressive run in which we made meaningful changes to the overall pipeline and how much each addition contributed to the final result.

Table 4.4: Ablation on **UCF-Crime** (each row toggles a single factor; training budget and optimiser settings are otherwise identical)

Variant	ROC-AUC $\uparrow$	PR-AUC $\uparrow$
<i>B1</i> CLIP+TSF, concat, $\gamma=1$ (no DPC)	46.2 %	22.3 %
<i>B2</i> + DPC predictive loss ($\gamma=0.5$, $\alpha=0.8$)	51.8 %	39.6 %
<i>B3</i> B2 + re-balance $\alpha=0.2$	57.9 %	47.5 %
<i>B4</i> B3 + Residual MLP Projection (1024 $\rightarrow$ 512)	70.4 %	61.1 %
<i>B5</i> B4 + LN-Gate (β residual) (our final)	84.5 %	72.3 %

- **B1: CLIP + TSF, concat, $\gamma = 1$ (no DPC)** This variant serves as the baseline. It utilizes CLIP (Contrastive Language-Image Pre-training, a common base for multimodal tasks) and TSF (Temporal Self-Attention/Features). Here, the term concat refers to combining the features through concatenation, which are extracted from CLIP and TimeSformer. The notation “ $\gamma = 1$ (no DPC)” implies a specific configuration of a parameter where DPC (a variant of Contrastive Predictive Coding extended to the video domain) remains inactive. Its performance, with ROC-AUC at 46.2% and PR-AUC at 22.3%, establishes a starting point for optimization.
- **B2: + DPC predictive loss ($\gamma = 0.5, \alpha = 0.8$)** Building based on B1, this variant introduces or activates the DPC predictive loss, making the inclusion of a predictive component aimed at enhancing temporal understanding or anomaly prediction. Specific hyperparameters for this loss are provided: “ $\gamma = 0.5$ ” and “ $\alpha = 0.8$ ”. The main advantage of DPC predictive loss is improving performance. After activating this, ROC-AUC has risen to 51.8 % and PR-AUC has risen to 39.6 %. The substantial increase in PR-AUC is particularly noteworthy, as this metric is often more sensitive to the imbalanced datasets typical of anomaly detection tasks.
- **B3: B2 + re-balance $\alpha = 0.2$** This variant enhances B2 by introducing a re-balance mechanism with a parameter “ $\alpha = 0.2$ ”. It involves an adjustment to mitigate the effects of class imbalance, a critical challenge in anomaly detection where anomalous events are rare. The re-balancing could be implemented through adjusted loss weights or specific sampling strategies. This modification further improves both metrics, with ROC-AUC that reach to 57.9% and PR-AUC to 47.5%, indicating better handling of rare anomalous events.
- **B4: B3 + Residual MLP Projection (1024 $\rightarrow$ 512)** This variant advances from B3 by including a residual MLP Projection. This denotes the addition

of a Multi-Layer Perceptron (MLP) with a residual connection, designed to project features from a dimension of 1024 down to 512. This step serves for dimensionality reduction, feature refinement, and increasing the capacity of the model to learn intricate relationships within the features. The inclusion of residual connections typically aids in the stable training of deeper networks. This addition results in a very significant performance boost, with ROC-AUC jumping to 70.4% and PR-AUC to 61.1%, suggesting effective refinement of the learned representations.

- **B5: B4 + LN-Gate (β residual) (our final)** This is the last phase that is considered as our most advanced variant, built on B4. It integrates two key components, where one is the LN-Gate (Layer Normalization Gate). It refers to a mechanism that controls information flow through a layer, potentially enhancing model stability and learning efficiency. Another is β Residual, which is a special type of residual connection integrated with the LN-Gate. It used to allow better gradient flow or feature refinement. This variant is explicitly highlighted as our final model. It achieves the highest performance among all evaluated variants, with a strong ROC-AUC of **84.5%** and PR-AUC of **72.3%**, demonstrating the effectiveness of the LN-Gate in providing a substantial final boost to anomaly detection capabilities.

Key observations: The key observation from the ablation process highlights the architectural improvements of our proposed model. It demonstrates how two major components - Residual MLP Projection and LN-Gate helped to boost anomaly detection performance incrementally. Each subsequent variant (from B1 to B5) shows a consistent and significant improvement in both ROC-AUC and PR-AUC scores, with the final B5 model achieving remarkable gains of **+38.3% in ROC-AUC (from 46.2% to 84.5%)** and **+50.0% in PR-AUC (from 22.3% to 72.3%)** over the initial baseline B1. This systematic improvement across variants highlights the importance of tailored components for refining features and optimizing information flow in complex anomaly detection tasks.

Chapter 5

Conclusion

In conclusion, this thesis introduced a context-aware zero-shot anomaly detection framework for surveillance system that integrates contrastive and predictive spatiotemporal modeling. It bridges the gap between low-level pattern anomaly detection and high-level situational awareness. Leveraging contextual information to tailor anomaly detection for scene-specific conditions, the proposed method integrates contrastive representation learning with predictive normalcy modeling to detect anomalies without the need for any training examples of abnormal data. Experimental results demonstrated that this method achieves high detection accuracy while maintaining a low false-positive rate. It effectively generalizes to previously unseen anomalous events across various scenarios. These findings highlight the potential of the proposed framework for real-world surveillance applications, as its modular, context-aware design allows for adaptation to new environments. While our approach has made notable progress, it is not without limitations. The dependency on context data means the system’s performance can degrade if such data are missing or incorrect. Also, extremely subtle anomalies or those involving complex interactions remain challenging. Future work has been proposed to address several issues, including improving context auto-discovery, enriching context modalities, enabling the model to explain anomalies in natural language, and refining the model for better efficiency and adaptability.

Bibliography

- [1] Tay, N. C., Tee, C., Ong, T. S., & Teh, P. S. (2019). Abnormal Behavior Recognition using CNN-LSTM with Attention Mechanism. 2019 1st International Conference on Electrical, Control and Instrumentation Engineering (ICECIE). <https://doi.org/10.1109/icecie47765.2019.8974824>
- [2] Van Den Oord, A., Li, Y., & Vinyals, O. (2018). Representation Learning with Contrastive Predictive Coding. arXiv (Cornell University). <https://doi.org/10.48550/arxiv.1807.03748>
- [3] Bertasius, G., Wang, H., & Torresani, L. (2021). Is Space-Time Attention all you need for video understanding? arXiv (Cornell University). <https://doi.org/10.48550/arxiv.2102.05095>
- [4] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. arXiv (Cornell University). <https://doi.org/10.48550/arxiv.2103.00020>
- [5] Cheng, Y., Wen, G., Luo, A., Mei, S., Dong, H., & Liu, X. (2025). An efficient and scale-aware zero-shot industrial anomaly detection technique based on optimized CLIP. *Measurement*, 117443. <https://doi.org/10.1016/j.measurement.2025.117443>
- [6] Marwa Qaraqe, Yang, Y. D., Varghese, E. B., Basaran, E., & Elzein, A. (2024). Crowd behavior detection: leveraging video swin transformer for crowd size and violence level analysis. *Applied Intelligence*, 54(21), 10709–10730. <https://doi.org/10.1007/s10489-024-05775-6>
- [7] Yang, C., Lan, S., Huang, W., Wang, W., Liu, G., Yang, H., Ma, W., & Li, P. (2022). A Transformer-Based GAN for Anomaly Detection. *Lecture Notes in Computer Science*, 345–357. https://doi.org/10.1007/978-3-031-15931-2_29
- [8] Aslam, N., & Kolekar, M. H. (2024). TransGANomaly: Transformer based Generative Adversarial Network for Video Anomaly Detection. *Journal of Visual Communication and Image Representation*, 100, 104108. <https://doi.org/10.1016/j.jvcir.2024.104108>
- [9] Ilyas, A., & Bawany, N. (2024). Crowd dynamics analysis and behavior recognition in surveillance videos based on deep learning. *Multimedia Tools and Applications*. <https://doi.org/10.1007/s11042-024-20161-7>

- [10] Deshpande, K., Narinder Singh Punj, Sanjay Kumar Sonbhadra, & Agarwal, S. (2023). Anomaly Detection in Surveillance Videos Using Transformer Based Attention Model. *Communications in Computer and Information Science*, 199–211. https://doi.org/10.1007/978-981-99-1648-1_17
- [11] Nasaruddin, N., Muchtar, K., Afdhal, A., & Dwiyantoro, A. P. J. (2020). Deep anomaly detection through visual attention in surveillance videos. *Journal of Big Data*, 7(1). <https://doi.org/10.1186/s40537-020-00365-y>
- [12] Esan, D. O., Owolawi, P. A., & Tu, C. (2020). Anomalous Detection System in Crowded Environment using Deep Learning. *IEEE Xplore*. <https://doi.org/10.1109/CSCI51800.2020.00012>
- [13] Sultani, W., Chen, C., & Shah, M. (2018). Real-world Anomaly Detection in Surveillance Videos. *arXiv*. <https://arxiv.org/abs/1801.04264>
- [14] Mahdyar Ravanbakhsh, Nabi, M., Enver Sangineto, Marcenaro, L., Regazzoni, C. S., & Sebe, N. (2017). Abnormal event detection in videos using generative adversarial nets. *International Conference on Image Processing*. <https://doi.org/10.1109/icip.2017.8296547>
- [15] Liu, X., Luo, W., Du, J., Wang, X., Dang, Y., & Liu, Y. (2024). A robust generalized zero-shot learning method with attribute prototype and discriminative attention mechanism. *Electronics*, 13(18), 3751. <https://doi.org/10.3390/electronics13183751>
- [16] Adín Ramírez Rivera, Khan, A., Ibrahim, E., & Taimoor Shakeel Sheikh. (2022). Anomaly Detection Based on Zero-Shot Outlier Synthesis and Hierarchical Feature Distillation. *IEEE Transactions on Neural Networks and Learning Systems*, 33(1), 281–291. <https://doi.org/10.1109/tnnls.2020.3027667>
- [17] Zhu, J., Cai, S., Deng, F., & Wu, J. (2024). Do LLMs understand visual anomalies? Uncovering LLM capabilities in zero-shot anomaly detection. *arXiv*. <https://doi.org/10.48550/arxiv.2404.09654>
- [18] Gu, Z., Zhu, B., Zhu, G., Chen, Y., Li, H., Tang, M., & Wang, J. (2024). FILO: Zero-Shot Anomaly Detection by Fine-Grained Description and High-Quality Localization. *arXiv*. <https://doi.org/10.48550/arxiv.2404.13671>
- [19] Jeong, J., Zou, Y., Kim, T., Zhang, D., Ravichandran, A., & Dabeer, O. (2023). WinCLIP: Zero-/Few-Shot Anomaly Classification and Segmentation. *arXiv*. <https://doi.org/10.48550/arxiv.2303.14814>
- [20] Li, A., Qiu, C., Kloft, M., Smyth, P., Rudolph, M., & Mandt, S. (2023). Zero-Shot anomaly detection via batch normalization. *arXiv*. <https://doi.org/10.48550/arxiv.2302.07849>
- [21] Li, Y., David, A. G., Liu, F., & Foo, C. (2024). PromptAD: Zero-shot Anomaly Detection using Text Prompts. *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 1082–1091. <https://doi.org/10.1109/wacv57701.2024.00113>

- [22] Zhou, Q., Pang, G., Tian, Y., He, S., & Chen, J. (2023). Anomaly-CLIP: Object-agnostic prompt learning for zero-shot anomaly detection. arXiv. <https://doi.org/10.48550/arxiv.2310.18961>
- [23] Gong, D., Liu, L., Le, V., Saha, B., Mansour, M. R., Venkatesh, S., & Anton. (2019). Memorizing Normality to Detect Anomaly: Memory-augmented Deep Autoencoder for Unsupervised Anomaly Detection. arXiv. <https://doi.org/10.48550/arxiv.1904.02639>
- [24] Hasan, M., Choi, J., Neumann, J., Roy-Chowdhury, A. K., & Davis, L. S. (2016). Learning Temporal Regularity in Video Sequences. arXiv. <https://doi.org/10.48550/arxiv.1604.04574>
- [25] Joo, H. K., Vo, K., Yamazaki, K., & Le, N. (2022). CLIP-TSA: CLIP-Assisted Temporal Self-Attention for Weakly-Supervised Video Anomaly Detection. arXiv. <https://doi.org/10.48550/arxiv.2212.05136>
- [26] B Ravi Kiran, Dilip Mathew Thomas, & Ranjith Parakkal. (2018). An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. arXiv. <https://doi.org/10.48550/arxiv.1801.03149>
- [27] Luo, W., Liu, W., & Gao, S. (2017). A Revisit of Sparse Coding Based Anomaly Detection in Stacked RNN Framework. 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, pp. 341-349. <https://doi.org/10.1109/ICCV.2017.45>
- [28] Pang, G., Shen, C., Cao, L., & Hengel, A. van den. (2020). Deep Learning for Anomaly Detection: A Review. arXiv. <https://arxiv.org/abs/2007.02500>
- [29] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021b). Learning transferable visual models from natural language supervision. arXiv. <https://doi.org/10.48550/arxiv.2103.00020>
- [30] Sun, S., & Gong, X. (2023). Hierarchical semantic contrast for scene-aware video anomaly detection. arXiv. <https://doi.org/10.48550/arxiv.2303.13051>
- [31] Wu, P., Zhou, X., Pang, G., Zhou, L., Yan, Q., Wang, P., & Zhang, Y. (2023). VADCLIP: Adapting Vision-Language Models for Weakly Supervised Video Anomaly Detection. arXiv. <https://doi.org/10.48550/arxiv.2308.11681>
- [32] Yang, Z., & Radke, R. (2024). Context-aware video anomaly detection in Long-Term datasets. arXiv. <https://doi.org/10.48550/arxiv.2404.07887>
- [33] Unusual Crowd Activity Detection in Video Using CNN, LSTM and OpenCV. (2023). Ijrasnet.com. <https://www.ijrasnet.com/research-paper/unusual-crowd-activity-detection-in-video-using-cnn-lstm-and-opencv>
- [34] Sharma, S., Sudharsan, B., Naraharisetti, S., Trehan, V., & Jayavel, K. (2021). A fully integrated violence detection system using CNN and LSTM. *International Journal of Electrical and Computer Engineering*, (IJECE), 11(4), 3374. <https://doi.org/10.11591/ijece.v11i4.pp3374-3380>