# Beyond Observation: The Role of Visual Question Answering in CCTV Footage Analysis

by

Towfiq karim
23241106
MD Tashin Rahman
23241071
MD Alvi Rahman
24141211
Minhaj Uddin
24141175
Mirza Bushra Tabassum
24141174

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

Department of Computer Science and Engineering
Brac University
October 2025

# Declaration

It is hereby declared that

1. The thesis submitted is my/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:**

Towfiq karim
23241106

MD Tashin Rahman
23241071

MD Alvi Rahman
24141211

Minhaj Udddin
24141175

Mirza Bushra Tabassum
24141174

# Approval

The thesis/project titled "Beyond Observation: The role of Visual Question Answering in CCTV footage analysis" submitted by

1. Towfiq Karim (23241106)
2. MD Tashin Rahman (23241071)
3. MD Alvi Rahman (24141211)
4. Minhaj Uddin (24141175)
5. Mirza Bushra Tabassum (24141174)

Of Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on October 10, 2025.

**Examining Committee:**

Supervisor:
(Member)

Md. Golam Rabiul Alam
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

Thesis Coordinator:
(Member)

Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

# Abstract

Visual Question Answering (VQA) system that revolutionizes CCTV surveillance through intelligent anomaly detection and automated incident reporting. The security and surveillance functions of CCTV cameras intensively capture terabytes of data everyday. An effective and efficient method should exist to extract footage and analyze its contents. Traditional methods find it challenging to work with high-dimensional along with complex data while require long period of time and being designed for specific tasks. Deep learning models in artificial intelligence have improved research analysis functionality by making operations more efficient. Visual Question Answering (VQA) relies on Natural language processing together with computer vision to produce its operations. Our research addresses this challenge by developing an integrated framework that combines advanced computer vision with natural language processing to enable real-time, query-based video analysis and automated security reporting. An innovative CCTV surveillance system based on VQA technology and build an unified system of the combination of state-of-the-art models of computer vision, TimeSformer, UniFormer, MotionFormer, and SlowFast, and natural language processing, BLIP-2, BART, OpenCLIP, and InstructBLIP, to operate in real-time to analyze the video and provide automated feedback about the detected anomalies through the query input. TimeSformer in general and TimeSformer with spatio-temporal dynamics in particular are shown to perform better with regard to capturing spatio-temporal dynamics and are 65% accurate in determining an anomaly on our dataset. AI-powered text generation allows the system to generate rich, context-wise summaries and answers, which make it much easier to interpret and use. The experimental findings indicate that the framework is successful in the management of low-resolution video data and noisy video data, which improves the efficiency and accuracy of analysis procedure in real-time video analysis. The work offers a significant background to smart and flexible surveillance systems that can conduct proactive surveillance and accurately conduct an anomaly in the sophisticated setting.



**Keywords:Visual Question Answering (VQA), Computer Vision, Natural Language Processing, CCTV Surveillance, TimeSformer, UniFormer, MotionFormer, SlowFast, Anomaly Detection, Text Generation, BLIP-2, BART, OpenCLIP, InstructBLIP, Adaptive Machine Learning, Real-Time Video Analysis.**

# Acknowledgement

We acknowledge the accomplishment of this research paper and owe all to Almighty Allah, who provided us the opportunity to study and explore. We owe our supervisor Dr. Golam Rabiul Alam or our co-supervisor MD. Tanzim Reza our great gratitude, for their valuable assistance and support throughout the study. We extend this gratitude further to all the lecturers and professors of Brac University who helped us improve the quality of our research. Finally, we would like to end by thanking our beloved parents for their support and positive encouragement.

# Table of Contents

**Declaration** i

**Approval** ii

**Abstract** iv

**Acknowledgment** v

**Table of Contents** vi

**1 Introduction** 1
- 1.1 Research Problem . . . 2
- 1.2 Research Objectives . . . 2
- 1.3 Report Organization . . . 3

**2 Literature Review** 4
- 2.1 Related works . . . 4

**3 Requirements, Impacts and Constraints** 17
- 3.1 Final Specifications and Requirements . . . 17
  - 3.1.1 Dataset requirements . . . 17
  - 3.1.2 Preprocessing requirements . . . 17
  - 3.1.3 Modeling requirements . . . 18
  - 3.1.4 Training and optimization requirements . . . 18
  - 3.1.5 Inference and outputs . . . 18
  - 3.1.6 Methodological specifications . . . 19
- 3.2 Societal Impact . . . 19
- 3.3 Environmental Impact . . . 19
- 3.4 Ethical Issues . . . 20
- 3.5 Project Management Plan . . . 21
- 3.6 Risk Management . . . 21
- 3.7 Economic Analysis . . . 22

**4 Methodology** 24
- 4.1 Dataset . . . 24
- 4.2 Data Pre-processing . . . 24
- 4.3 Model Specification . . . 26
  - 4.3.1 TimeSformer vs Uniformer . . . 26
  - 4.3.2 Uniformer Implementation . . . 27
  - 4.3.3 TimeSformer vs MotionFormer . . . 28

4.3.4 MotionFormer Implementation: . . . 30
4.3.5 TimeSformer vs SlowFast . . . 32
4.3.6 SlowFast Implementation: . . . 33
4.3.7 TimeSformer Model Architecture . . . 34
4.3.8 BLIP-2 Model Architecture(Bootstrapping Language-Image Pretraining) . . . 36
4.3.9 BART Model Architecture . . . 38
4.3.10 InstructBLIP Model Architecture . . . 41
4.3.11 OpenCLIP Model Architecture . . . 44
4.4 Model Pipeline . . . 47

**5 Result and Discussion** **56**
5.1 Performance Metrics Description . . . 56
5.2 Query Retrival Performance Matrix: . . . 60

**6 Limitations and Future Work** **64**

**7 Conclusion** **68**

**Bibliography** **71**

# Chapter 1

# Introduction

The model known as Visual Question Answering or VQA created around 2010 has not only become popular but also contributed to the association of natural language processing and computer vision. By integrating VQA technology, robots are capable of analyzing a textual question about visual content and providing an answer; this way, robot language comprehension is combined with picture analysis. The challenges of VQA have evolved since the early use of the model to simply identify objects and describe their attributes. It uses a real method as a recurrent neural network today to handle queries and a convolutional neural network to handle images. These developments have extended their uses from just research work to offer practical solutions in other areas, such as security systems, where they enhance the functionality of surveillance, and medical diagnosis, where the functionality of the visual data is enhanced. As a consequence, VQA is one of the most important technologies that can change the way people interact with the digital environment as conversational interfaces develop and become more sophisticated.

The real-world difficulties that security professionals frequently have while dealing with the huge amount of CCTV footage that needs to be inspected every day serve as the inspiration for this study. Currently, it takes a lot of time to navigate this large amount of film in order to find specific events. This has the disadvantage of time delay in replies in the organization since it is not only slow but also prone to human errors. This research aims to modify this process to enable the user to describe an event in natural language by using Visual Question Answering (VQA) technology so that the system can identify the right video. It has the opportunity to significantly reduce an index of the workload of security teams and shift their efforts toward higher value activities and increase the productivity and efficiency of the security processes.

The revolutionary capability of Visual Question Answering to alter human interactions with surveillance systems forms the basis of our study. Imagine a situation in which security personnel can communicate with their monitoring systems in the same way as they would normally ask a colleague for information. Our idea is to make surveillance operations more logical and effective by developing a VQA that will be able to understand complex questions and identify the fragments of the video containing the answer. Advanced monitoring technologies could become more accessible to low-funded organizations through this approach leading to broader im-

provement of security measures worldwide. This research is driven by a need to do better and safer, and the goal of this research is to come up with a more intelligent and people-oriented way of analyzing CCTV footage.

The conventional CCTV analysis employs operators who observe video footage manually; thus, it is slow, erroneous, and appropriate for limited investigations. Visual Question Answering (VQA) solves the task of interpreting a video feed on its own, and a viable solution is possible by combining the technologies used in image processing, such as object detection and facial recognition. VQA takes surveillance beyond its conventional role of monitoring since it transforms visuals into valuable intelligence. This enables behavior tracking, pattern detection, and complicated question-answering regarding CCTV footage, providing precise footage of an occurrence as reported. This breakthrough would especially prove most beneficial within situations that demand high-level security, urban monitoring, and crime prevention. Therefore, due to the integration of CCTV's system, the operation of the security becomes more effective because more complete and faster access to information is provided, the action that looks more suspicious is identified more accurately, and therefore, better decisions are made.

## 1.1 Research Problem

Despite advances in surveillance technology, most CCTV monitoring systems still depend on manual review of video, leading to slow and error-prone incident detection, especially when timely response is crucial. While modern automatic solutions can detect basic objects or events, they often cannot interpret the natural language questions or detailed search requests that users make. Existing systems lack the ability to accurately match user queries with the relevant segments in large, unconstrained video datasets, limiting their usefulness in real investigations.
Therefore, the core research problem is to develop a robust Visual Question Answering (VQA) system that enables users to ask complex, natural language questions about surveillance footage and receive direct, relevant video responses. The system must address challenges related to understanding multi-step, contextual queries, scaling to large databases, and reducing the human effort needed to monitor and review surveillance videos, by bringing together video analysis, natural language processing, and efficient retrieval techniques. The goal is to make surveillance video search faster, more reliable, and responsive to real-world needs.

## 1.2 Research Objectives

This research aims to build a better system for answering questions about events captured by CCTV cameras. By using natural language processing, face recognition, object detection, and machine learning, the system can search and find video clips that match the descriptions people provide in everyday language. This approach is designed to make it much easier for users to quickly retrieve the CCTV footage they need by simply describing the incident they are interested in. The objectives of this research are:

1. Develop an integrated Visual Question Answering (VQA) and anomaly detection framework tailored for large-scale CCTV surveillance video, enabling natural language search and reasoning over multi-class anomalous events.

2. Design and implement novel multi-stage NLP strategies to accurately interpret and decompose user queries—including multi-action, relational, and temporal questions—for intelligent segment retrieval and context-aware video analysis.

3. Combine video models (TimesFormer, BLIP-2, OpenCLIP) with captioning and summarization so each search result shows what happened in the video and why it may be important, making it easier for users to understand the results.

4. Engineer a scalable retrieval and indexing system based on multimodal semantic embedding (OpenCLIP+FAISS), enabling efficient real-time search and anomaly localization across tens of thousands of video shots in surveillance-scale deployments.

5. Create and validate a highly usable, interactive interface that allows users to submit written or spoken language queries, receive compositional, context-preserved answers with visual evidence, and iteratively refine search intent for high-stakes monitoring scenarios.

6. Systematically evaluate semi-supervised and unsupervised learning strategies to improve generalization for unseen anomaly types, particularly for rare or evolving incidents that lack annotated samples, thereby bridging the supervision gap common in real-world surveillance VQA.

## 1.3 Report Organization

The rest of this paper is arranged as follows. Chapter 2 narrates a wide literature review which explore into the existing literatures which are related to VQA along with their progress, limitations and future scope in these sector. Chapter 3 presents Methodology along with input data pre-processing, model training and the workflow diagram and also presents the proposed model architecture. Chapter 4 includes the Results and Discussion and lastly Chapter 5 concludes the paper.

# Chapter 2

# Literature Review

## 2.1 Related works

Visual Question Answering(VQA) is an emerging field of modern Artificial Intelligence,Computer Vision, and Natural Language Processing that involves answering questions based on visual image or videos.

The evolution of VQA proposed by Stanislaw Antol et al.(2016)[2] where AI systems provide answers to questions that describe the action or occurrence in an image or video and one needs to understand why to understand and answer such questions. The authors generated a dataset of more than Two Hundred and Twenty thousand images and Seven hundred and sixty thousand questions with 10 million responses from the MS COCO dataset and some abstract scenarios. They have trained a baseline using CNN for image feature extraction and LSTM for the processing of the questions. The operation of these models involved the combination of image and question in one feature space and simple text-based answers. Although for simple queries, the performance of the models was impressive and above expectations, the results which emerged when asked more challenging questions that involved reasoning and even more basic stuff like applying commonsense knowledge exposed the model's weakness such as dataset bias, lack of external knowledge, and poor generalization. This paper has laid a foundation on which future works in multimodal Artificial Intelligence could be built on however, in terms of reasoning and external knowledge the VQA systems should be enhanced for it to perform at the level of human intelligence.

In [14] the D-VQA method has verified samples that are helpful in dispensing biases specific to VQA when cross-distributions are trained and tested. But because of a complex construction of the model and because the choice of the negative instances is crucial, the effectiveness of the model is rather confined and its application is rather difficult to extend. However, the model architectural structure seems complicated and as a result incurs high computational complexity and hence the model cannot be implemented for real large scale problems in real world problems. Furthermore, the condition that the negative examples must be chosen when necessary also decreases the number of possibilities for using the model for multiple tasks and areas. Future work should focus on how to decrease the complexity of the models in the hope of decreasing the computing cost as well as keeping a high efficiency on bias

mitigation. This may entail coming up with new samples to follow training frameworks that would not call for many resources and minimize the bias. Furthermore, the automation or simplification of the process of generating the negative samples might actually turn out to enhance the versatility of the model when used in different fields and tasks within the VQA domain and activity, as well as its accessibility.

Yang et al.(2021) [16] proposed a novel dataset HowToVQA69M, a large number of question-answer pairs for videos, yet there are significant differences related to the reliance on transcribed, noisy annotations of the videos. From the influences, speech recognition and a mismatch between the questions and films' real substance are the reason for these discrepancies. As a result, the noise in the data disrupts the process of model learning and, therefore, reduces performance as well as generalizability, especially in those activity types that imply a strong match in the questions and the visual content. It is about time that voice recognition increases its capacity coupled with more effective question creating approaches in a bid to reduce these issues on question-answer compatibility. From this, better model learning results would be achieved since some of the dataset noise would be reduced. The same would also increase the diversity and generalization capacities of the models which are trained on the given set and the reach of the inclusion might go farther to include instructional, documentary, and user-generated videos.

Although VQA and video analysis work has some problems, especially for complex tasks that involve temporal, causal and relational understanding, In [15] refers that VQA models trained with NExT-QA dataset perform well in descriptive context understanding part of VQA while fails in relating temporal and causal connection between events in video footage. A major limitation of these models is that, for instance, it is not easy to describe in detail an action sequence of agents, and often it is a problem to predict the result or cause and effect to explain what happened. In particular, when posing temporal questions to the past, present, or the future, existing models fail to perform thinking abilities beyond description. More complex models that reflect temporal and causal framework should supply the need. It might involve examining in what contexts the more recent kinds of models based on graphs can be used to describe the dependencies between actions in time and maybe underlining the idea that one event leads to another event. Such models should primarily focus on temporal action justification, so they are able to reason not only on the content level-"what", but also on the time and purpose of actions happening.

In [12] the discussion about Graph Interaction Networks moves on to their misalignment with dynamic content in videos. However, whereas networks capable of representing fixed visual data work, more complex relationships between appearance, motion and language within video sequences can not be well represented. Current models are not fully understanding temporal dynamics of video, such as the movements of objects, interactions and transitions, creating challenges in answering complex open-ended questions. Future work should also build on the temporal dynamics processing in graph interaction networks by modeling long term dependencies and mixing better the static and dynamic information. In this case, better bridging mechanisms might be developed between appearance and motion representations to enable the model to process and understand sequential and interrelated

events that take place in a video more effectively.

Difei et al. (2020) [8] introduced Multi-modal graph neural networks (MM-GNNs), work has demonstrated that a range of modalities, including visual, textual and numerical can be integrated in Multi-modes into tasks including scene text recognition and visual quality assessment. However, the solution of such models requires incredibly high computational time and resources to train such models, meaning they cannot be applied to real-time contexts or large datasets. Moreover, relative to the features of raw data, it is challenging to understand the decision making process, which is particularly problematic in highly nuanced fields such as medicine where, in particular, the logic of a modeled decision is crucial to confidence and security. The subsequent research can be directed on how to enhance the process of creating the graphs in order to reduce the complexity of the computation either by increasing the efficiency of the approach or by approximating it. Furthermore, the application of explainable AI (XAI) methods, MM-GNNs may become more interpretable, which enhances their capability in obtaining a better understanding and building trust to the extent they will be employed in the real world.

Additional elaboration of the concept of data quality and model performance. In [21] we discuss specific issues that arise in the surveillance application of VideoQA: occlusions, limited fields of view, and lighting conditions interfere with their ability to answer questions based on the video content. Also, the kinds of questions present in the dataset are not very massive and sophisticated to challenge the model to perform complicated tasks like multiple object tracking or recognition, activity and behavioral analysis across the cameras. To justify the dataset, deeper dataset augmentation including the variation of camera perspectives, better lighting conditions, and advanced questions' types would contribute to the model's calibration. Further improvement of the feature extraction process to work with the occlusions or inclusion of more contextual information such as audio, metadata or environmental information would greatly improve the performance of the model in real-world video surveillance scenarios.

Lui et al. (2020) [11] proposed HAIR framework, which shows its remarkable precision of relational reasoning, but has challenges of modeling complex and implicit semantic relationships, primarily in longer video sequences. In addition, sampled in the real world of just 10 frames per video the model fails to capture key temporal dynamics especially in long or more complex video sequences and thus fails to capture the understanding of actions and events. Complex situations demand more fine grained temporal and spatial dynamics to be captured by the model. Such models can also be used to improve performance involving higher semantic interpretation and temporal reasoning such tasks by adding other modalities-audio or speech-to give a more integrated picture of the video information.

Jiyang G. et al.(2018) [4] in their Motion-Appearance Co-Memory Networks (MACN) model contribute to video question answering by making good use of motion and appearance information. Since the primary discussed goal of the research aims at lifting the capabilities of the model to the reasoning of the temporal structures in videos, the dual issues of visual features capture (appearance), the dynamics of

actions (motion) and the adaptive behavior of the model to different kinds of questions are addressed. The paper suggests a Motion Appearance Co Memory Network (MACN) that makes use of co memory attention mechanism to allow motion and appearance features interaction flow to control attention generation. Moreover, it uses dynamic fact ensemble method to livelily integrate the required temporal data on each question, involving a multi-level temporal convolution-deconvolution network to extract contextual facts on video frames. We test this model on the TGIF-QA dataset, a general videoQA benchmark which also includes repetition counting, action recognition, state transitions and frame QA. Even though the MACN model demonstrates significant advances in VideoQA, it continues to have problems with learning out of complex multimodal data and can possibly be assisted with the addition of a more diverse range of real world examples and lower level temporal reasoning to the memory update scheme.

Yushen et al. (2023) [25], have describe the Visual Causal Scene Refinement (VCSR) framework that leverages a variety of video content understanding tasks such as causal reasoning, visual scene sensing. VCSR further improves video QA system's accuracy and interpretability through the Question Guides Refiner and Casual Scene Separator to refine video segments and separate causal scenes. Although the framework beats present methods with strong results on causal question answering particularly, it is constrained by increased computational demands, relying on detailed annotations and more broadly, by potential problems in generalizing to unseen data. In the future, extending the dataset diversity, reducing computational overhead, improving the generalization capability and integrating the unsupervised learning techniques into the VCSR framework can lead to making the VCSR framework more robust and applicable in various real life scenarios.

The journey then pushes in **[11]** to explore novel methods which leverage the advancement of deep learning technologies to improve VQA capabilities. The research seeks to increase the range of VQA to the level of how machines understand and respond to enlightened natural language questions about the images and videos. We take a step towards the ultimate goal to impart human-like understanding and reasoning ability in machines, by taking VQA as the stepping stone. Technically, the work is based on the recent models such as Co-Memory Networks and Hierarchical Relational Attention mechanisms and combines them with Generative Adversarial Networks (GANs) to learn unsupervised. We present the SVideoQA dataset, which is specifically constructed on the context of surveillance videos and which is designed to address the issue of multiple camera feeds of video. According to the results of the evaluation, these approaches are the best compared to the conventional state-of-the-art approaches in terms of video sequence comprehension and multi camera configurations, with huge improvements in monitoring systems. In spite of this the thesis in fact acknowledges limitations when dealing with high dynamic situations and proposes future direction of research for unsupervised learning techniques and adaptive reasoning capabilities. This work's contribution is to make the VQA system more robust and able to run in practical systems, most importantly in ensuring security and monitoring systems are significant.

Maitreya et al.(2022) [19] have introduced the CRIPP-VQA dataset which entails

counterfactual arguments on physical properties in videos. The dataset uses counterfactual reasoning and planning task to challenge models to comprehend both visible(e.g, shape,color) and invisible(e.g, mass,friction) characteristics of objects in video. It has more than 4,000 videos which investigate the effects of such actions as removing, replacing or adding objects on the physical effects, showing the capability of the model to think about their physical effects. The results of benchmarking indicate that the model such as Aloe+ can answer approximately 71 per cent of descriptive questions about visible properties, but fail to perform well in counterfactual reasoning and planning task where accuracy reduces significantly in out-of-distribution (OOD) situations where the unknown object properties are observed. Shortcomings of using existing models are absence of object detection in the face of complicated surroundings as well as the issues in the proper depiction of the physical interaction such as mass or friction variations. Lastly, the paper suggests that work in the future focus may be made to be more robust to these implicit physical properties and more challenging reasoning tasks.

Andrew Shin 'et al in their paper [5] have proposed a novel system for generating personalized image descriptions by including user interactions through visual question generation and answering. The authors have identified a major problem with conventional image captioning methods, which typically produced static and generic descriptions that failed to account for individual user interests. To solve this problem, they propose an interactive model where users are asked different questions about different regions of an image. Based on their answers, the system learns from the answers and knows about their preferences and generates more customized and interactive narratives. This approach takes the system even deeper into factual one size fits all descriptions, and sews the story to fit the user's interests. For image captioning the model uses datasets such as MS COCO and for generating questions related to the image it helps use the VQG dataset. The results show that it is significantly improving over the baseline models like DenseCap and SIND in more diverse, interesting and more expressing narratives. Insite being a really good model, the paper highlights few areas for future research and development, such as generalization of user preference across more images, expanding the complexity of questions and integrating reinforcement learning to continuously adapt the system based on ongoing feedback.

In [18] the authors introduce SwapMix, a technique that improves the stability of VQA models by preventing them from attending to unrelated visual contexts. However, the analysis of results also reveals that SwapMix has limitations in solving complex relational reasoning problems, especially if a problem being solved implies relations between objects and actions that need to be described in detail. The current model suffers from the innate inability to represent and naturally reason about these relational dependencies. Future work could extend models such as SwapMix through better methodologies focusing on object and action dependencies in relational reasoning. Moreover, the increased generation of irrelevant object swaps for training purposes augmenting the diversity of perturbations to the model could be task independent and thus, also increase the model's performance of other tasks and its resistance in various or in more difficult situations.

Following the tradition of innovating in user interaction in [28], an advanced anomaly detection model called BiTCN_MHA, which is a combination of a Bidirectional temporal Convolutional Network (BiTCN) and a Multi-Head Attention (MHA) mechanism is presented to drastically extend the performance of anomaly detection in time series data. It overcomes the limitations of existing statistical methods as well as modern deep learning by utilizing both forward as well as reverse temporal features during data analysis. MHA is introduced to the model, which further prioritizes critical features like improving the model and helps in preventing information overload as well as overcomes the neuron death problem seen in ReLU activations using the ELU activation function. Moreover, BiTCN_MHA is superior in consistency to state with of the art models on many applications ranging from healthcare to server monitoring. However, the model is also plagued by excessive computational complexity and its reliance on quality and diversity of training data thus it is difficult to implement the model in resource constrained environments and indicates a trade off between performance and operational ability in real world applications.

Doyup et al. (2020) [9] have proposed a new way of enhancing anomaly detection in VQA models through attention based method. Incorporating this method provides a far superior confidence metric compared to conventional softmax based confidence metrics like Maximum Softmax Probability(MSP). We test this method against a variety of VQA models on data sets such as VAQ v2,MINIST,CIFAT-10 and Tiny ImageNet and focus on out-of-distribution(OOD) images and questions as well as unanswered pairs. The proposed anomaly detection technique based on attention significantly improves anomaly detection performance while causing minor deterioration in overall VQA accuracy. Nonetheless, there is a decrease in accuracy, particularly after applying Outlier Exposure(OE), which is up to 7.7%. The paper also proposes further improvements on anomaly detection with regularization to use maximum entropy and suggest the scope for robustness and reliability enhancements within VQA systems, which are pivotal for real world application where accuracy of anomaly detection is necessary for its safety and effectiveness.

In [6] further extrapolating this story of anomaly detection and model robustness to detect anomalies in real world surveillance videos with the use of a new technique on them, which makes the process of labeling them a lot easier as it can be done at a video level, rather than at an image level, MIL (Multiple instance learning) would be used. Nonetheless, it is merely a methodology of treating the entire video as bags and the pieces of the videos as instances, which significantly decreases the amount of work done on the annotation and makes it more scalable. There are the sparsity and the time-smoothness constraints that are exploited to train the models of anomaly ranking that learn to differentiate between the positive (anomalous) and negative (normal) video bags, guaranteeing both detection performance and stability. It allows accurate localization of short sporadic anomalies that are of paramount interest in a surveillance environment. Lastly, the paper introduces a novel and extensive collection of 1900 real-world videos containing numerous realistic anomalies, which is publicly available to be used in the creation of the anomaly detection technology field. The model experiences certain difficulties in dark surroundings and blocked scenes. The false positives are reduced, however, and anomalies are correctly identified, indicating the points of future improvement of the work with complex visual

dynamics.

In [1] address the major issue of false alarm in surveillance systems, which are repeatedly being triggered by various environmental conditions such as crowd movements and change in lighting. To solve this issue, authors came up with a reduced-reduction (RR) approach to maximize computational efficiency and detection accuracy by using fewer reference features than the traditional approaches, yet better than no-reference approaches. It relies on edge energy and gray-level standard deviation as some of the important image characteristics and uses and online Kalman filter that effectively removes noise and minimizes false positives associated with temporary environmental variations. The effectiveness of this approach was confirmed with a specially designed collection of observations of surveillance videos (only 75), including various cases of both deliberate and unintended camera defects. It is clear that the proposed solution demonstrates greatly superior results obtaining the highest levels of performance and accuracy with high recall measures while keeping the false alarm ratio at a less level. This positive result implies a striking improvement over the prior techniques. However, there is still improvement as the robustness of the system for complex noises from the environment is severe and in various scenarios such as light changes and camera vibrations, so that the performance of the anomaly detection in more scenarios in real complex settings can be improved.

Dewashee et al. (2020) reviewed [13] the incorporation of edge computing to address the challenges in anomaly detection for video surveillance. Authors mention the increasing challenges of manually running large volumes of surveillance data and the shortcomings of the existing cloud-based solutions, the latter being affected by bandwidth and latency concerns in real-time systems that demand fast responses such as anomaly detection. To address these issues, the authors consider using edge computing with the cloud systems to allow real time anomaly detection to process data near the source. They have surveyed different deep learning methods. As an example, CNNs, LSTMs and GANs to detect anomalies, segmenting algorithms depending on targets such as individuals, crowds and automobile vehicles. As for training, such traditional datasets as UCSD and Avenues are used for comparing the effectiveness of the models for identifying anomalies in surveillance systems. These datasets are used as standard sets to measure high performance accuracy and computation methodology of deep learning models in controlled environments. The results demonstrate that edge computing can further enhance the current system to decrease latency and enhance the real-time anomaly detection capability. However, some authors have pointed out the following limitations that should be addressed. Furthermore limitations that have been identified by some authors are. These are the current shortages of 'clever' algorithms that would perform well in limited devices, also additional improvement of coordination between edge devices and the cloud system for efficient management of tasks. Future works are the following: Optimizing real time processing using adaptive models and extending the functionality of edge computing to address more complex scenarios such as crowded environments and multi camera setups.

Difel Gao et al. [17] evaluates the Long-form Video Question Answering (VideoQA) and faces challenges from high computational expenses for video processing and

limited capabilities of tracking multiple event connections and specific video elements due to sparse sampling. Such video assessment requires models to face challenges with their complexity and their long-term temporal nature. By implementing the Multi-Modal Iterative Spatial-Temporal Transformer the MIST model solves both these issues. MIST uses cascaded segment and region selection module to determine the most significant video frames and image regions depending on the question. The computational overhead is reduces and the self-attention mechanism functions through successive layers to improve understanding and unite different temporal periods and measurement scales in its analysis. The rational reasoning system performs multiple event analysis while dealing with fine-grained visual concepts through a series of iterative steps. MIST delivers the best performance on four VideoQA benchmarks including AGQA and NExT-QA while surpassing previous models for multi-event reasoning and understanding causality relations and identifying object-action associations. The computational efficiency reaches superior levels when compared to conventional transformers due to MIST. VideoQA can be successfully addressed by the MIST approach when it comes to long content using its efficient accuracy control. The adaptive selection mechanism together with its iterative attention process allows it to process long video content leading innovations in Multi-Modal AI technology.

According to Zi-Yuan Hu, the existing Video LLMs demonstrate difficulty at understanding temporal information et al.[26], single-frame biases are normally employed. TG-Vid has a Time Gating (TG) module which enhances temporal modeling by allowing. the model adapts spatial and temporal attention and MLP layers with respect to input and output data. The TG-Vid model has been built on the Vicuna-7B LLM with EVA-ViT-g vision encoder and on InstructBLIP-based QFormer to train the model using video-text pair datasets. It performs better when compared to other Video LLMs in terms of performance on MVBench (56.4% accuracy) and TempCom pass (59.6% overall) and NExT-QA (67.5% on ATP-Hard, 77.3% on validation) on caption matching tasks. The ablation tests prove the functionality of the TG module particularly through its attention mechanisms of time sequence data. TG-Vid method discloses the relevance of explicit temporal attention to video under- standing through attaining the most developed performance. Future work that researchers should take into consideration is to introduce the limits of video duration, as well as introduce new video-and language evaluation areas.

This paper [20]Vision transformers (ViTs) represent revolutionary tools which compete against the traditional Convolutional Neural Networks (CNNs) to recognize actions. The paper breaks down Vision transformer based action recognition models through an analysis that includes their arguments and architectural designs as well as training methods and performance measurements. Two categories exist for action transformers including CNN-integrated where CNNs extract spatial features while transformers maintain temporal understanding and pure transformer models achieve spatial and temporal understanding through frame-based self-attention. Pure transformer models achieve better results than traditional CNNs though both depend on similar standard training parameters consisting of 16/32 frame input and 16x16 kernels through AdamOptimizer mechanisms. The combination of MTV with Kinetics400 reached 89.9% accuracy along with VideoMAE reaching 75.4% ac-

curacy in SSV2. At the same time Epic-Kinetics-100 achieved 53.6% with MM-Mix and HD-GCN scored 90.1% performance on NTU-RGB+D 120. Although Vision transformers solve many problems they encounter operational challenges which include limited motion understanding efficiency and a slow processing speed. Future work in this field will involve transformer optimization for better design and motion detection enhancement through optical flow analysis and combines multiple inputs with self-supervised system training practices.

Damith C.S et al. [27] introduces CUE-Net as a new deep learning framework for automated violence detection that uses spatial cropping along with Enhanced UniformerV2 architecture as well as Modified Efficient Additive Attention. Surveillance system deployments have risen so high that manual human monitoring became impractical which led to the need for automated solutions. The solution developed by CUE-Net incorporates spatial cropping with an enhanced UniformerV2 architecture and a Modified Efficient Additive Attention (MEAA) mechanism which optimizes violent activity detection in video contents. Solar CUE-Net operates with five modules which possess: YOLO V8-based spatial cropping for frame selection and people detection and 3D convolution and local-global UniBlocks that utilize MEAA alongside Multi-Head relation aggregators followed by a fusion block for classification. CUE-Net architecture is implemented in PyTorch with AdamW as its optimizer and cosine learning rate scheduling to train the model and could apply all benefits of CLIP-ViT embeddings to enhance features extraction. The model demonstrates cutting-edge performance on RWF-2000 and RLVS benchmark datasets by achieving 94.00% accuracy and 99.50% accuracy respectively outperforming the Video Swin Transformer along with ACTION-VST and DeVTr and any other CNN-based and ViT-based model variations. The performance of CUE-Net increases 0.5% through spatial cropping and reaches 1.5% more accuracy with MEAA replacing Self-Attention in the Global UniBlock while decreasing GPU memory usage from 47.33GB to 35.04GB. The model has good visualization results in misclassification exercises in the detection of intricate violent behavior but indicates a bit of uncertainty in the boundary of vague cases. The research shows that CUE-Net apply local features extracted with convolution algorithms and attention based global context analysis while it is operating efficiently producing a highly effective system for real-time surveillance for violence detection applicable to public safety.

The UniFormer framework presented in this paper[24] brings together CNNs with ViTs to create an architecture which achieves visual recognition accuracy while maintaining efficient performance.CNN perform better on local features efficiently but limits their ability to process in distant relationships. ViTs manage extensive relations but causes heavy computational power. In UniFormer processes starts by applying the convolution like operations in shallow layers for reducing redundancy before moving on to self attention in deeper layers to capute global dependencies. The model merges both Dynamic Position Embedding with Multi-Head Relation Aggregation and Feed Forward Network as components that enables effective lock feature extraction with global feature learning. Extensive experiment reveals that the model delivers better result across all visula recognition task.UniFormer demonstrates superior performance on ImageNet-1K by reaching 86.3% top-1 accuracy while consuming less resources than both CNN-based and ViT-based models. Top-1

accuracy on Kinetics-400 and Something-Something V2, of 82.9% and 71.3% respectively, demonstrates that the model has strong temporal modeling. AAP of the COCO objects detected was 50.3 AP and segmentation AP was 44.8 AP with UniFormer compared to the results on Swin Transformer. A reduced version of UniFormer has a 2-4 times better throughput which is essential to real-time applications and does not reduce its performance. Unlimited transformer employs combined approaches of local and global feature extraction which exceeds numerous previous approaches while minimizing duplicate calculations. UniFormer shows promising real-world capabilities since it can be applied to autonomous systems as well as surveillance and medical imaging.

In their paper "Where to Look: Focus Regions for Visual Question Answering" Kevin J. Shih, Saurabh Singh and Derek Hoiem[3] present a model which learns to determine image focus points according to given questions during Visual Question Answering. Current VQA processing models analyze entire images but waste resources since they have to process the whole image when only a particular region needs attention. The model proposed converts the questions and the image pieces to the same feature spaces in order to quantify their meaning correlation in a selective way of seeing significant areas. In the given approach, word2vec text embeddings are used along with CNN region features within a mechanism of attention to prioritise various parts of the image. The region selection model demonstrates superior performance on the VQA dataset and obtains remarkable results in localizing tasks because of its hinge loss training method that assigns higher scores to correct answers. Through its implementation the model demonstrates higher accuracy at 58.94% exceeding average accuracy of whole-image-based models (57.83%) and uniform region averaging (57.88%). The proposed model generates superior outcomes than base methods when detecting colors (53.96% versus 43.52%) yet demonstrates strong ability in scene categorization (86.21% against 76.65%) and displays limited development for visual counting and text understanding activities. Learning in In VQA word2vec model performs much better than LSTMs since its dynamic focus adjustment mechanism generates better accuracy with interpretability. Integration of pre-trained detectors and geometric reasoning mechanisms as well as external sources knowledge could be used to enhance the methodology. The visual question-answering methodology receives a crucial enhancement in terms of the fact that Where to Look that makes image region scrutiny highly effective in enhancing question-answer results.

When the convolutional features are inadequate to encode the long-range dependencies in surveillance video, Biradar et al. (2024) proposed TEF-VAD (Transformer Encoded Feature Video Anomaly Detection) to overcome the challenge of identifying subtle anomalies in surveillance video. The strategy substitutes convolutional backbones with stacking multi-head self-attention layers, which learns a variety of spatio-temporal representations. The model is trained on three specialized losses, including feature magnitude learning, class-specific contrastive, and Multiple Instance Learning classifier loss which direct the model to focus on anomaly cues, distinction between normal/abnormal embeddings, and incorporate clip-level features into video-level scores. TEF-VAD reported 88.3% frame-level AUC on UCF-Crime (4.7% higher) and 85.9% AUC on ShanghaiTech (5.2% higher), which is the new state-of-

the-art performance.

Kainulainen et al. (2024) [30]introduced MM-Diff-Net the initial diffusion-based multimodal video captioning model that attempts to address the drawbacks of autoregressive decoders to achieve cross-modal coherence. The model combines the frames of video, audio waveforms, and speech transcripts to a single step of the latent diffusion. An attention-based multimodal feature aligner A noise-conditioned U-Net sequentially iteratively removes noise in multimodal embeddings, and multi-modal attention modules show the similarity of features between modalities. A modality-sensitive scheduler regulates steps of diffusion by modality to balance generation. MM-Diff-Net had an 112.5 CIDEr score on YouCook2 (or +9.4), 29.8 METEOR on MSR-VTt (or +3.7), and 31.2 BLEU-4 on VATEX (or +5.1) outperforming transformer baselines on benchmarks.

Hu et al. (2024)[**27**] address temporal reasoning failures in Video LLMs with TG-Vid, that adds a Time Gating module to an LVLM pipeline that uses Vicuna-7B LLM, EVA-ViT-g vision encoder, and InstructBLIP Q-Former. Time Gating layer Time-varying token embeddings of frame timestamps are dynamically modulated by the model to give a higher weight to recent frames and a lower weight to long-term ones. TG-Vid scored 56.4%, 59.6% and 67.5% on MVBench ( +9.2 over baseline), TempCompass ( +7.8) and NExT-QA ATP-Hard ( +12.1) respectively, which was a significant improvement in temporal understanding.

Senadeera et al. (2024)[**28**] presented CUE-Net as a method of automatic violence detection in surveillance, given the infeasibility of human surveillance in large-scale CCTV networks. YOLOv8 is used in the pipeline to crop people and regions, 3D convolutions are used to capture short-term motion features and an Enhanced UniformerV2 backbone is used to combine the local UniBlocks that capture detailed spatial information and global UniBlocks that capture cross-patch attention. A Modified Efficient Additive Attention module improves time aggregation, and a focal loss classifier deals with the imbalance in classes. CUE-Net scored 94.00% on RWF-2000 ( +3.5 over Video Swin) and 99.50% on RLVS ( +2.1 over CNN-based methods) and reduced the consumption of GPU memory by 35.0 GB versus 47.3 GB.

Zheng et al. (2025)[29] launched SurveillanceVQA-589K, the biggest open-ended VQA benchmark in the surveillance area, to fill the gap of large-scale QA dataset on CCTV applications. The dataset includes 589,380 pairs of questions and answers of 12 question types and 18 categories of abnormal events that were produced through a hybrid gas pipeline GPT-4 and human annotation model. Some of the question types such as, descriptive (What is happening?), causal (Why did the alarm sound?), and temporal (When did the person leave?). The benchmarking of eight LVLMs showed a median accuracy of 42.7% on descriptive, 35.4% on causal and 28.9% on temporal queries, and found that VQA abilities in terms of surveillance are considerably differentiated.

Li et al. (2024)[**32**] introduced LS-VIT, a Vision Transformer with Long and Short-term Temporal Difference Modules to enhance the ability of videos to model tasks of motion understanding. The Short-Term, and Long-Term modules calculate the pixel

change between frames to get instantaneous motion edges and features respectively, then temporal pooling is learned respectively. These tokens are particles of motion injected into ViT self-attention blocks, which makes it possible to jointly learn space-temporal without using 3D convolutions. LS-VIT performed with 94.2% on UCF101 (+4.4 over baseline ViT) 80.5% on HMDB51 (+5.2) and 79.1% on Kinetics-400 (+5.5) showing that the model can temporal model effectively.

Wang and Li (2025)[**31**] develop a vision transformer enhanced with LBP to identify abnormalities in surveillance by combining Local Binary Pattern texture feature and deep ViT feature. Patch embeddings are gated multimodal attention after embedding LBP histograms and concatenating them. The cross-entropy and center-loss are used in training to guarantee the condense intra-class features. The cross-entropy and center-loss are used in training to guarantee the condense intra-class features. The model attained 97.0% accuracy on abuse, 95.2% on fighting, and 94.4% on robbery with a custom surveillance dataset, which was 3-5 times the pur-ViT baselines.

Shi et al. (2024)[**30**] study attention processes in video diffusion models with the IE-Adapt method and their results find that high-entropy attention maps are related to video quality and low-entropy attention maps are related to structural fidelity. IE-Adapt is a dynamic cross attention scaling system that scales the cross attention using map entropy, enhancing the underutilized attention channels and reducing the noisy ones. IE-Adapt when used on a base video diffusion model increased Fréchet Video Distance by 12% on UCF-101 to Vimeo-90K and SSIM by 8% on DAVIS, confirming the usefulness of entropy-based attention adaptation.

Lewis et al. (2019)[7] proposed a new denoising sequence-to-sequence pre-training architecture, BART (Bidirectional and Auto-Regressive Transformers). BART operates by corrupting input text with noising techniques that are flexible (span infilling (randomly masking spans of text)) and sentence permutation) and learns to restore the original text. BART is very adaptable to both generation and understanding tasks because it uses a bidirectional encoder (such as BERT) and an autoregressive decoder (such as GPT). BART produced state of the art scores on various abstractive generative tasks. It, e.g., improved on the XSum summarization dataset by an impressive +6 ROUGE points, surpassed RoBERTa on GLUE and SQuAD general language understanding tasks, and showed better results in ELI5 (abstractive QA) tasks. Also, BART has seen a +1.1 BLEU score gain on the WMT16 English-Romanian machine translation task. This performance on diverse natural language processing (NLP) tasks that are performed well indicates that BART is able to transfer knowledge across domains (generation and comprehension).

Bertasius et al. (2021)[10] suggested TimeSformer, a video understanding model that makes use of pure Transformer architecture to classify video. TimeSformer proposes a space-time attention division mechanism, in which space and time attention are factorized, which greatly reduces computational efficiency. This mechanism of attention separates the space-time interaction making it possible to efficiently process long video clips and high-resolution inputs. On large scale video classification tasks, such as Kinetics-400, Kinetics-600, Something-Something-V2, and Diving-48, Timesformer was found to be a competitive state of the art (SOTA)

model. To be precise, TimeSformer had a Top-1 accuracy of 80.7% on Kinetics-400 using its large variant (L). Learning space-time positional embeddings also increase the effectiveness of the model in comparison to space-only models, providing a more comprehensive view of space and time of video data. The effective utilization of computational resources offered by TimeSformer allows it to become a beneficial model to use in video recognition tasks, which are considered as large-scale, and it proves that Transformers can be used to process spatio-temporal data with a comparable level of accuracy and efficiency.

Li et al. (2023)[23] proposed a new advanced Vision-Language Model (VLM) BLIP-2 (Bootstrapping Language-Image Pretraining): frozen image encoders and huge pre-trained language models (LLMs) are connected by a small Querying Transformer (Q-Former). BLIP-2 is trained in two phases: (i) representation learning which trains the model to learn combined image-text representations with attention-masking techniques to improve the alignment of the model vision and language output, and (ii) vision-to-language generation during which the model is trained to learn to generate textual descriptions of query embeddings as soft visual prompts into an encoder-only or encoder-decoder LLM. Such an architecture enables BLIP-2 to execute a large variety of tasks in vision-language including visual question answering (VQA), image captioning, and visual grounding with high performance. The model was also found to be 8.7% better in zero-shot tasks than Flamingo, another SOTA VLM, with 54 times less trainable parameters. This finding supports the claim that BLIP-2 can perform well at significantly reduced computation needs and, thus, can be an effective choice in the area of vision-language large-scale tasks.

Dai et al.(2023)[22] introduced InstructBLIP which is a modification of BLIP-2 making it instruction-tuned to make its performance on an extensive variety of vision-language tasks. The InstructBLIP is optimized to be able to operate with instruction-aware Q-Former that conditions the visual feature extraction of the model with textual instructions. This innovation enables the model to be more understanding and adaptable to a range of tasks through use of instructions as opposed to only through multitask learning. The model was experimentally tested on 26 datasets of 11 task categories, and it was demonstrated that multitask formatting was not the most important factor in improvement of zero-shot generalization to unseen datasets, rather instruction tuning. InstructBLIP, also, registered new SOTA on a number of benchmarks, including ScienceQA, with an 90.7% accuracy on FlanT5-XXL. The model was also efficient in that the visual encoder was frozen and the model only needed about 188 million trainable parameters. The good result of instructBLIP in zero-shot tasks and efficient utilization of model parameters make it a significant addition to the understanding of vision-language learning, especially to tasks involving instruction-following skills.

# Chapter 3

# Requirements, Impacts and Constraints

## 3.1 Final Specifications and Requirements

- The system integrates Visual Question Answering with video anomaly detection for CCTV, centering on TimeSformer for classification and pairing vision models with language models for captioning, summarization, and instruction-following.
- Vision models compared and/or incorporated include TimeSformer (primary), UniFormer, MotionFormer, and SlowFast; language and multimodal components include BLIP-2, BART, OpenCLIP, and InstructBLIP.
- Retrieval and indexing are designed around multimodal semantic embeddings and FAISS (OpenCLIP+FAISS) to enable scalable query-time search across large shot collections.

### 3.1.1 Dataset requirements

- Source: "Real-world Anomaly Detection in Surveillance Videos" (UCF-Crime), organized as Anomaly-Videos-Part-1 to Part-4, with normal video parts for training and annotated anomaly start/end frames for testing.
- Target label space (example mapping used in implementation): Abuse → 0, Arrest → 1, Arson → 2, Burglary → 3, Explosion → 4, Burglary → 5, Normal → 6, Road Accident → 7, Robbery → 8, Shooting → 9.
- Data splits and sampling: frames extracted at 5 fps; an 80/20 train–test split is used in the main preprocessing description, while a 70/30 split is specified in the MotionFormer implementation section; class folders contain roughly 45 videos per class.

### 3.1.2 Preprocessing requirements

- Frame extraction from each video into class-named directories, RGB conversion for grayscale frames, resizing to 224×224, and normalization to either [0,1] or [-1,1].

- Foreground masking with SAM2 to emphasize object-, human-, and action-centric regions and suppress background noise.
- Optical flow generation with S-Flow to capture dense motion cues (velocity, direction, temporal continuity) between consecutive frames.
- Shot construction via a sliding window of 3 seconds with a 1.5-second stride; per-shot sampling of 16 frames, middle-frame keyframe selection, and storage of shot metadata and labels in `shots.jsonl`.

### 3.1.3 Modeling requirements

- Primary classifier: TimeSformer trained on two-stream fused inputs (RGB frames for appearance and optical-flow frames for motion dynamics) to predict anomaly class probabilities at the shot level.
- Captioning: BLIP-2 applied to per-shot keyframes to generate semantic captions; BART and InstructBLIP architectures are described as language/backbone components for generation and instruction-following.
- Retrieval: Multimodal embeddings via OpenCLIP with FAISS indexing are specified in the research objectives for real-time search and anomaly localization across large shot collections.
- Alternative backbones evaluated: UniFormer, MotionFormer (dual-stream RGB+flow), and SlowFast (dual temporal resolution paths), with fine-tuning protocols covering head-only, partial, and full-model phases.

### 3.1.4 Training and optimization requirements

- Frameworks and tooling: Python and PyTorch implementations with custom training loops and device-aware optimizations.
- Optimizer and schedules: AdamW optimizer with StepLR learning-rate scheduling; dropout and label smoothing for regularization.
- Stability and efficiency: mixed-precision (autocast and GradScaler), gradient clipping, and class weighting to handle class imbalance.
- Data augmentation: RandomResizedCrop, horizontal flip, color jitter, and Gaussian blur to improve generalization.

### 3.1.5 Inference and outputs

- Shot-level classification uses softmax over the 10 anomaly classes to produce probabilities and confidence scores per shot, driven by fused RGB and optical-flow inputs.
- Outputs include clip-level logits, timeline anomaly scores for localization, spatiotemporal attention maps for interpretability, and description relevance scores for incident matching.

- Scene captions are generated for keyframes via BLIP-2; low-confidence cases can trigger reprocessing with refined SAM2 masks, longer clip windows, or higher resolution to improve feature extraction.

### 3.1.6 Methodological specifications

- End-to-end pipeline: data ingestion → preprocessing (RGB/flow/masks) → shot extraction and metadata → two-stream fusion → TimeSformer classification → BLIP-2 captioning → retrieval and incident matching via embeddings and indexing.
- Validation and checkpointing rely on held-out performance of clip-level anomaly classification consistent with the thesis' description of model selection and evaluation procedures.

## 3.2 Societal Impact

This thesis suggests a retrieval system of surveillance which can positively enhance the workflow of the community in terms of safety and cause a decrease in the load of the operation. The system reduces investigative lead times, facilitating a quicker emergency response, and reducing human error that is common in manual video viewing by allowing the use of natural-language search through large CCTV archives and presentation of evidence-ready shots with captions, summaries, and model confidence. Accessibility of a question-answer style interface reduces the skill barrier of small or resource-limited organizations (e.g., schools, transit hubs, clinics) and increases the socio-technical coverage of the state of the art video analytics. In addition to incident triage, the summarized results (ranked results, anomaly patterns and summaries) can be used to inform prevention strategies and urban planning based on data driven information. Simultaneously, I acknowledge and manage privacy, fairness, and transparency risks: I describe guardrails like purpose-limited use, access control and audit logging, retention policies, bias-conscious assessment, and human-in-the-loop verification of high-stakes decisions. Combined, the contributions will aim at transforming CCTV into an active storage into explainable and accountable decision support, capable of providing practical benefits in safety and efficiency whilst not violating civil liberties.

## 3.3 Environmental Impact

As in this example, I evaluate the environmental performance and sustainability of the proposed CCTV retrieval system under the actual work conditions. Given that the system involves reuse of the available camera infrastructure, hardware production is also reduced and e-wastes are minimized. The most critical factors leading to the predominant footprint and hardware life-cycle factors are the calculation (training/fine-tuning), data storage/transfer. I would model these "carbon hotspots" through efficiency-first (quantization, mixed precision, early-exit routing), temporality (right-sized fixed clip length fixed frame rates), and edge/near-edge processing (only transmits keyframes, summaries, embeddings). Other energy-related metrics, embodied emissions reduction, are carbon conscious scheduling,

extreme data minimization (short TTLs, compression, deduplication), checkpoint-reuse, and hardware stewardship. My KPIs are (kWh and kgCO2e, joules per analyzed hour/query, storage TB and retention compliance, network ingress/egress, edge share) and demand accuracy/latency re-validation of any change that is 10% energy saving. Tiered storage and pruning, early stopping of model runs with energy spikes, and certified recycling mitigate storage bloat, energy spikes, and e-waste. Overall, operational advantages of the project with a small and auditable footprint and indirect advantages (e.g. less idling and congestion of emergency vehicles and faster incident localization) are achieved by allowing faster incident localization.

## 3.4 Ethical Issues

# Key ethical issues

- **Privacy & consent:** Non-consenting subjects can be filmed/ analyzed; the threat of sensitive information leakage.
- **Purpose creep:** The tool designed to triage incidents may be used to spy on people in large numbers.
- **Data minimization:** Data collected by video/embeddings should be minimized since it enhances breach and misuse of data.
- **Bias & fairness:** The different error rates in different contexts/demographics may be unequally harmful.
- **Explainability:** black box scores/labels are counterproductive to review and challenge.
- **Security:** Model artifacts, keys and indexes have high value targets.
- **Accountability & compliance:** Must be traceable when auditing them; CCTV/PII/biometrics legislations can be relevant.

# Professional responsibilities

- **Privacy by design:** Edge processing, least data, short TTLs, role-based access.
- **Clear scope & policy:** Document acceptable use; prohibit tracking/profiling beyond lawful purpose.
- **Fairness audits:** Test subgroup performance; calibrate thresholds; add human review for high-risk cases.
- **Transparency:** Show keyframes/rationales, calibrated confidence, immutable audit logs; publish model card.
- **Security controls:** MFA, encryption, least privilege, regular pen-tests, breach response plan.

- **Lifecycle governance:** Versioning, drift monitoring, impact assessments (DPIA/AIA) at each update.
- **Legal alignment:** Map data flows to applicable laws; verify licenses; manage cross-border transfers.
- **Stakeholder engagement:** Train operators; provide signage/notices; enable complaint/redress channels.

## 3.5 Project Management Plan

We have three main phases for our thesis, starting with Pre-Thesis 1 followed by Pre-Thesis 2 and lastly the final defense. Our Pre-Thesis 1 started in Spring 2024 and lasted for around 4 months, the whole spring semester to be precise. This phase had formation of our thesis team, topic and supervisor selection of thesis domain,topic and supervisor, summarisation of related researches and works and finally the submission of the report. Pre-Thesis 2 was conducted in the Summer 2024 semester which also had a timeline of 4 months. In this phase, we worked on analyzing the data and building the translation models that we want to implement. After that, we submitted a report. Finally, in Spring 2025, we analyzed our results and compared them with existing works to prove the depth and accuracy of our work. Here, we completed our paper and will be facing our Thesis Defense in front of the qualified panels. The Gantt Chart is a visual representation of the whole Thesis timeline. The three main phases are shown in different colors for more clarity.

## 3.6 Risk Management

There are several possible practical and ethical risks in this project, to which we reduce the likelihood of occurrence through physical measures. To begin with, computational overhead and latency can impede time-sensitive application of the two-stream, transformer-based models, we overcome this by quantizing, pruning, mixed-precision directing early-exit routing and tiered pipeline to allow a lightweight model to be executed initially with only escalation applied on-demand. Second, capture at fine scale, or long time context can be lost; we are enhancing RGB-optical-flow fusion, and (where it can be useful) hybrid conv/transformer representations. Third, overfitting and poor generalization are risks due to the class and domain shift imbalance, we use more ardent augmentation and class-rebalance, watch high-quality validation indicators on held-out domains and use early-stopping and disciplined-ablations. Fourth, bad-image-quality or noisy images may propagate errors to captions and summaries, and we apply data-quality filters, optional denoising/super-resolution where available (where permitted) and also require human review where we are too confident about the model. Fifth, hardware training can be scarce, and this can be at the cost of accuracy, reuse of checkpoints, run fewer and longer, and scale capacity using the accumulation of gradients. Sixth, inaccurate ranking and improperly selected confidence can ruin the trust of the operators; operators optimize probabilities, separate keyframes/rationales, establish conservative setups and demand through the human-in-the-loop reviews during escalation. Seventh, storage and query traffic to a surveillance scale may strain indexing and throughput, we

have FAISS HNSW optimized in store policies, edge pre-filtering, strict retention / compaction policies and batch refresh. Finally, the privacy, mission creep and regulatory exposure are the high-impact risks, we are employing privacy-by-design (edge processing, minimal retention, role-based access), irrevocable audit records, model/version lineage and conspicuous acceptable-use policies that are monitored. Such checks are defined by governance decision gates (data readiness, model viability, E2E latency) in such a way that the failure to meet a decision gate is compensated by fall backs: the use of light backbones, the use of lower resolution/ clip length, the use of reduced pilot scope.

## 3.7 Economic Analysis

The economic side of this thesis project is based on the price of the computing resources, software infrastructure, and manpower needed to construct, train and test a large-scale AI-powered video surveillance and retrieval system. Because the system combines several deep learning models including TimeSformer, MotionFormer, SlowFast, BLIP-2, BART, and OpenCLIP. It took a high-performance computing installation and a few months a month of trial and error.

**Hardware and Resource Costs:** The whole pipeline was tested and made on a high-configuration workstation that has an NVIDIA RTX A6000 graphics card, an Intel Core i9 (current generation) processor, 128GB RAM and 2TB SSD storage. Such a setup costs the market around 8-9 lakh BDT, which is estimated to cost 7,000-8,000 USD. The system was run in the period of about twelve months in the model training, testing, and integration. The total energy cost of about 300-400 USD can be calculated by adding to the cost of the existing CPU power a GPU electricity consumption (about 400 W per hour) over an active computation time of about 1400 hours. The added cost of minor peripheral costs of backup drives, cloud storage of FAISS indexing, and local storage of datasets would be approximately an addition of $200 USD.

**Software and Tooling:** A majority of the development frameworks applied in the present project, such as -PyTorch, Transformers, FAISS, OpenCLIP, Streamlit, and Hugging Face libraries are open-source, which helped to reduce software costs. But in a large scale implementation player, the operation costs of cloud GPU servers would cost about **$300 -500 USD/month** of hosting and inference.

**Manpower and Development Time:** The project was done in a research cycle of 12 months, which consisted of continuous data collection, data preprocessing, model training, performance evaluation, and system design. Assuming the same rate of research assistant at 500 USD/month, then the effective manpower investment will be approximately 6,000 USD.

| **Component** | **Description** | **Approx. Cost (USD)** |
|---|---|---|
| Hardware Setup | RTX A6000 GPU, i9 CPU, 128 GB RAM, 2 TB SSD | 7,500 |
| Power and Maintenance | Electricity, cooling, and wear | 400 |
| Software and Storage | Open-source tools, data storage | 200 |
| Manpower | 12 months of research and development | 6,000 |
| **Total Estimated Cost** | | **14,100 ( 17 lakh BDT)** |

Table 3.1: Estimated Cost Breakdown for the Thesis Project

**Cost–Benefit Evaluation:** The development cost is relatively high because of the requirement of powerful hardware and lengthy training times but the benefits in the long term are far much better than the costs. The system suggested can save human resources significantly in the area of CCTV analysis where it can take days of manual analysis of the footage to save a few seconds in query-based retrieval. Upon deployment, the cost of a query is low as it will be retrieved based on pre-indexed embeddings and using an FAISS-based search that is memory- and compute-efficient. What is more, all the model stack can be reused within other application areas like forensic investigation, traffic control, or industrial safety, and therefore yields more payback.

**Economic Viability:** In a wider sense, such an AI-driven surveillance system in urban surveillance can reduce the time spent on manual analysis and reaction by 8090 per cent, which may save thousands of work hours per year in a law-enforcement or security agency. Scalability and sustainability is guaranteed by the system due to the use of open-source components and edge processing features without the need to pay the software license again. Therefore, even though the project will require initial investment, it is economically viable and would have high long-term value through the enhancement of security efficiency, decreased human fatigue, and allowing real-time and intelligent surveillance analytics.

# Chapter 4

# Methodology

## 4.1 Dataset

Our enhanced VQA solution uses the anomaly detection module based on a sequence-organized dataset initially introduced by Chen Chen in his research at UNC-Charlotte as reported in the article *Real-world Anomaly Detection in Surveillance Videos* (CVPR, 2018). This dataset comprises various video segments, specifically:

- **A**nomaly video data is available to the users in the form of the four zip files, which are called Anomaly-Videos-Part-1 through Part-4.
- **T**o develop the system of identifying typical patterns of behavior, the training process needs normal video parts organized in Part-1 and Part-2.
- **E**laborating on exact starting and finishing frames of abnormalities in testing videos.

## 4.2 Data Pre-processing

1. **Frame Extraction:**
    - Each video is converted into image frames.
    - The frames are stored in directories that are named after the class of the anomaly.
    - The approach enables models to derive spatial data from distinct video frames as a result.
2. **Dataset Structuring:**
    - The frames extracted are stored in folders with classes in a `train/` and a `test/` directories.
    - The video sequences are divided into folders with the same category.
    - The distribution divides the data into 80% training and 20% testing.
3. **Frame Normalization and Pre-processing:**

- Frames are transformed to 3 channels **(RGB)** in case any frame is detected as a grayscale frame.
- Frames are downscaled to **224×224** so that they are stable.
- In order to be numerically stable, pixel values must be brought between $[0, 1]$ or $[-1, 1]$.

4. **Data Loading and Label Encoding:**
   - A mapping is established between each category and a numerical label:
     - **Abuse** $\rightarrow 0$
     - **Arrest** $\rightarrow 1$
     - **Arson** $\rightarrow 2$
     - **Burglary** $\rightarrow 3$
     - **Explosion** $\rightarrow 4$
     - **Burglary** $\rightarrow 5$
     - **Normal** $\rightarrow 6$
     - **Road Accident** $\rightarrow 7$
     - **Robbery** $\rightarrow 8$
     - **Shooting** $\rightarrow 9$
   - The `VideoDataset` class is designed to load frames along with their labels for deep-learning models.

5. **Creation of Mask Dataset (Foreground Extraction):** In order to make the dataset optimal with regards to the use in the anomaly detection tasks, Segment Anything Model 2 (SAM2) was used to achieve accurate foreground segmentation. This operation successfully segregated the object-centric, human-centric, or action-relevant areas of the 3-D video frame and reduced the background visability. The aim of this step was to focus the model on semantically important entities and minimize variance brought in by nonrelevant environmental contexts. The masked data that was obtained were a more discriminative representation of the data in a visual form which was fit to analyze high-level activities.

6. **Motion Dynamics Modeling (Optical Flow Generation):** Upon extraction of the foregrounds, Separable Flow (S-Flow) was employed in estimating dense optical flow fields between two consecutive frames. The results of optical flow maps achieved provided cues of fine-grained motions, like velocity, direction, and time continuity of motions. Such patterns of motion representations played a key role in a model of temporal dynamics and of anomalies (rapid or irregular motion patterns). Introduction of optical flow allowed introduction of time-dimension to complement the spatial appearance features.


Figure 4.1: *RGB image to Masked image*

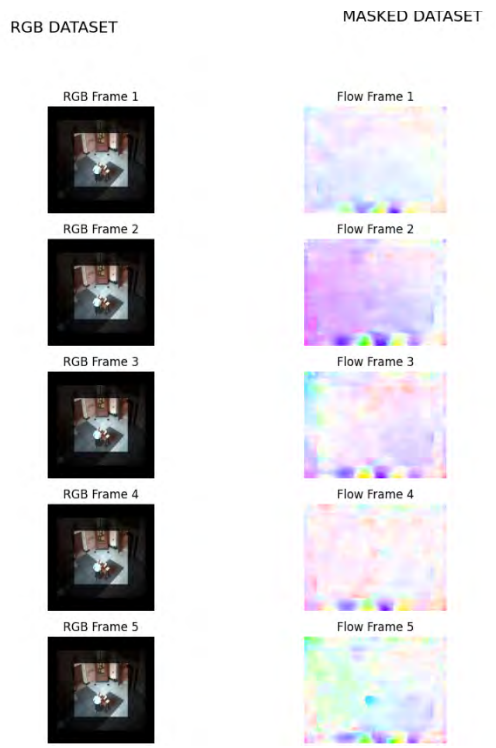


Figure 4.2: *Masked image to Optical image*

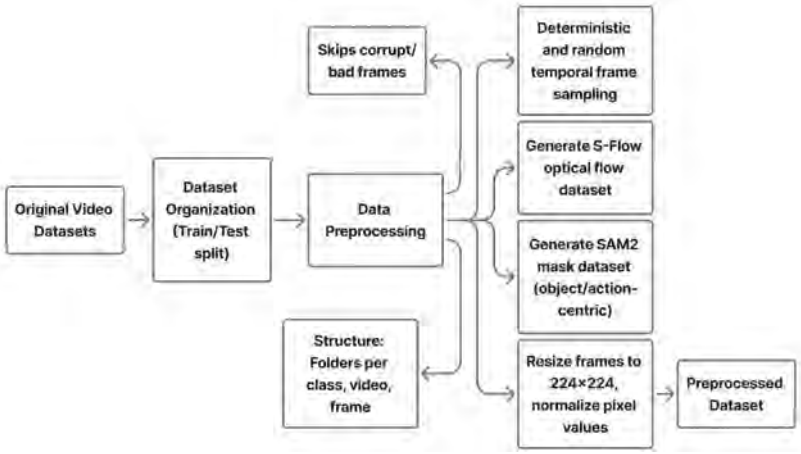


Figure 4.3: Data Pre-Processing

## 4.3 Model Specification

### 4.3.1 TimeSformer vs Uniformer

To create an optimal video-based anomaly detection system, we had to examine the TimeSformer and UniFormer that are advanced deep learning models. Here, we state the reasons why we have chosen the TimeSformer as our primary anomaly

detecting model and explain why we had to select the TimeSformer, as opposed to the UniFormer.The TimeSformer has demonstrated itself to be the best model in nearly all of the metrics that we have evaluated, specifically in terms of accuracy and ROC-AUC values. The selected metrics are important because they can be used to determine the degree of success of the model in differentiating between normal and anomalous video segments as well as the overall control of the model in different and complex video scenarios. In addition, TimeSformer can utilize well its divided space-time attention mechanism in processing video information. The system is very good to capture temporal dynamics with spatial information that introduces accuracy in the detecting of anomalies during the video surveillance tasks. Several in-depth tests and analysis of TimeSformer shows that it is suitable in our applications in intelligent surveillance systems due to the reliability of its working ability and accuracy under various work conditions. The performed analysis confirms why we chose TimeSformer to build and implement in our future anomaly detection applications.

#### Comparative Analysis

**Test Accuracy Over Epochs:** The initial demonstration indicates that TimeSformer performed better in test accuracy performance compared to UniFormer during all measurements periods. The TimeSformer had a test accuracy of approximately 60% and the UniFormer had difficulties in attaining a test accuracy better than 35%. The TimeSformer demonstrates outstanding capacity to transfer knowledge based on training data by its uninterrupted leader in performance in terms of accuracy.

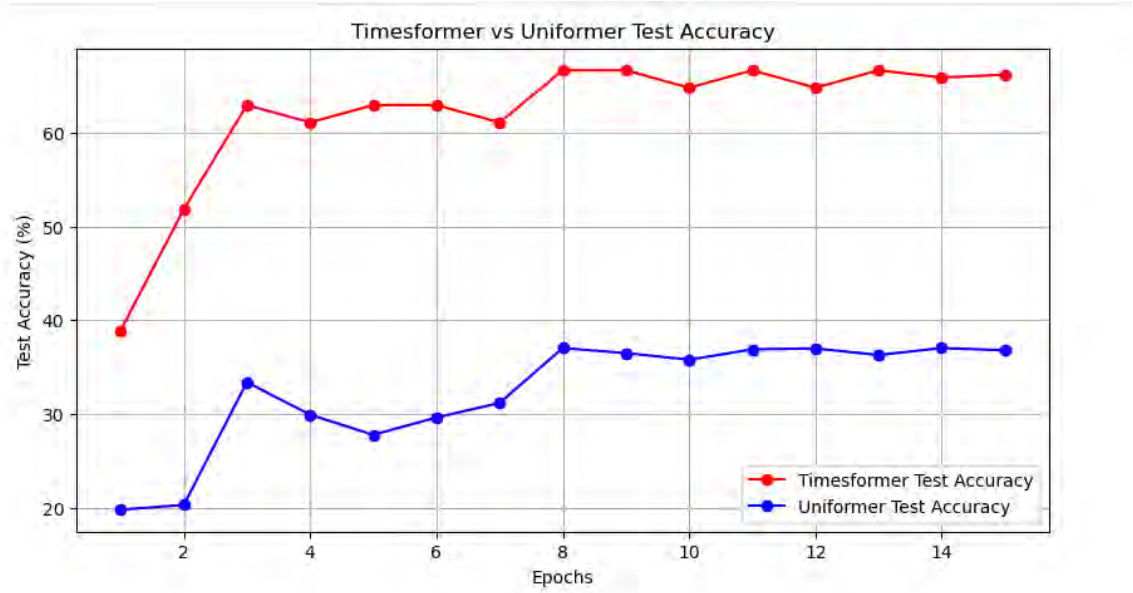


Figure 4.4: *TimeSformer VS Uniformer Test Accuracy*

### 4.3.2 Uniformer Implementation

This chapter presents the application approach along with empirical tests for the TimeSformer model that has been specifically designed to detect anomalies in video

surveillance data. Using theoretical constructs and preliminary methodologies from Thesis Part 1, this part of the research uses practical deep learning methods to develop video-based surveillance enhancements. The implementation follows a complex sequence of processes which performs data transformation on raw videos through TimeSformer model optimization with enhanced video knowledge.

**Data Pre-processing:** Advanced Data Processing Techniques: There are two sophisticated data processing techniques that allow researchers to extract frames correctly and label them in an intelligent way to generate good input material. It is a process that transforms Thesis Part 1 data optimization strategies to work with high-dimensional video data.

**Model Training and Fine-tuning:** Deep Learning Frameworks: TimeSformer works at this step once refined by more sophisticated methods deters temporal and spatial anomalies in video streams. The code operates with Python language while implementing PyTorch as well as other libraries to refine the training process through multiple error prevention systems along with improved device operations and enhanced data manipulation strategies.
The complete code implementation incorporates multiple sophisticated methods:

- Mixed Precision Training: The model is able to efficiently utilize autocast and GradScaler to optimize the use of the GPU-based training operation without compromising the precision.
- Gradient Clipping: Clips gradient explosion by restraining the value of the gradients.
- Class Weighting: The training process becomes more balanced through weight calculation which gives each class equal priority.
- Dynamic Learning Rate Adjustment: The StepLR scheduler allows the model to make periodic learning rate reduces that makes the model converge more easily.
- Data Augmentation: The model involves various transformations to enhance the ability of the model to generalize visual images across varying sources.
- Learning Rate:

$$\text{LR} = \text{Learning Rate} \times \min\left(\text{step}^{-0.5}, \text{step} \times \text{warmup\_steps}^{-1.5}\right) \tag{4.1}$$

- Dropout: Dropout layer normalization serves with a custom learning rate scheduler to train large-scale datasets using techniques which are vital for handling complex TranSformer dynamic learning mechanisms.

### 4.3.3 TimeSformer vs MotionFormer

TimeSformer, as well as MotionFormer, are strong models in the framework of video anomaly detection, but they both utilize the Transformer architecture. Although both models deal with video data, they are different regarding their ability to deal with temporal and spatial data. Below is a detailed comparison:

**TimeSformer:**
TimeSformer is a Vision Transformer (ViT) architecture that is directly applied to video data. Vision Transformers use a self-attention mechanism to learn image patch-patch relationship, and TimeSformer builds upon this idea to learn both space-dependent and time-dependent relationship in video.

- **Spatial Attention:** In TimeSformer, the frames are considered as a series of patches. The spatial attention process is concentrated on acquiring relationships of objects inside the frame (boundaries, textures, and shapes of objects). This enables the model to reproduce the appearance of objects in the video frames.

- **Temporal Attention:** The most significant innovation in TimeSformer is the introduction of temporal attention in order to process the time-order of video data. This process assists the model to acquire knowledge as to how various frames within the video are linked over time, have motion dynamics and alterations within the scene.

TimeSformer uses the spatial-temporal attention to ensure that the model captures finer details in a frame and across frames therefore it is very efficient in doing the anomaly detection where the model needs to understand both space (appearance) and time (motion).

**MotionFormer:**
MotionFormer is created with the special aim to cope with the problems of both appearance (spatial characteristics) and motion (temporal dynamics) in the video data. In contrast to TimeSformer, which independently applying spatial and temporal attention, MotionFormer is based on a dual-stream model, which takes two separate input streams, namely RGB (spatial) and optical flow (motion).

- **Dual-Stream Architecture:** In MotionFormer, RGB frames are used to record the surface of the scene (static objects, textures) and optical flow is used to record the motion dynamics (movement, object trajectories). The two streams are then handled independently by their corresponding encoders and then cross modal attention mechanism is utilized to join the two.

- **Motion Dynamics:** MotionFormer uses optical flow (dense motion features between successive frames) which is a major point of difference. Optical flow stream is especially useful in detecting anomalies that may concern fast or irregular motions, e.g. violent behavior, accidents, or anomalous behavior. The cross-modal attention mechanism means that the model is able to acquire the interaction between the spatial aspect and the motion aspect, and thus this model would be very appropriate in motion-related anomaly detection tasks.

MotionFormer focuses on motion, as its main attribute of anomaly detection, and therefore it is very successful when motion-based anomalies have to be detected. Its dependence on flow as well as RGB characteristics enables it to attain high performance when it comes to capturing complex dynamics of time.

**Comparative Analysis:**
The comparison of accuracy of TimeSformer and MotionFormer gives results that TimeSformer is much better than MotionFormer. During the initial 25 epochs, TimeSformer quickly reaches an accuracy of approximately 70% on the test, which proves to be a good learning and generalization. Conversely, MotionFormer has slower improvements, as the test accuracy remains about 20-25% throughout most of the epochs, and then there is a slight improvement. This indicates that TimeSformer is more efficient in the initial training phase without the need to address both spatial and temporal features whereas MotionFormer needs longer to optimize its dual-stream structure to motion-based anomaly detection.

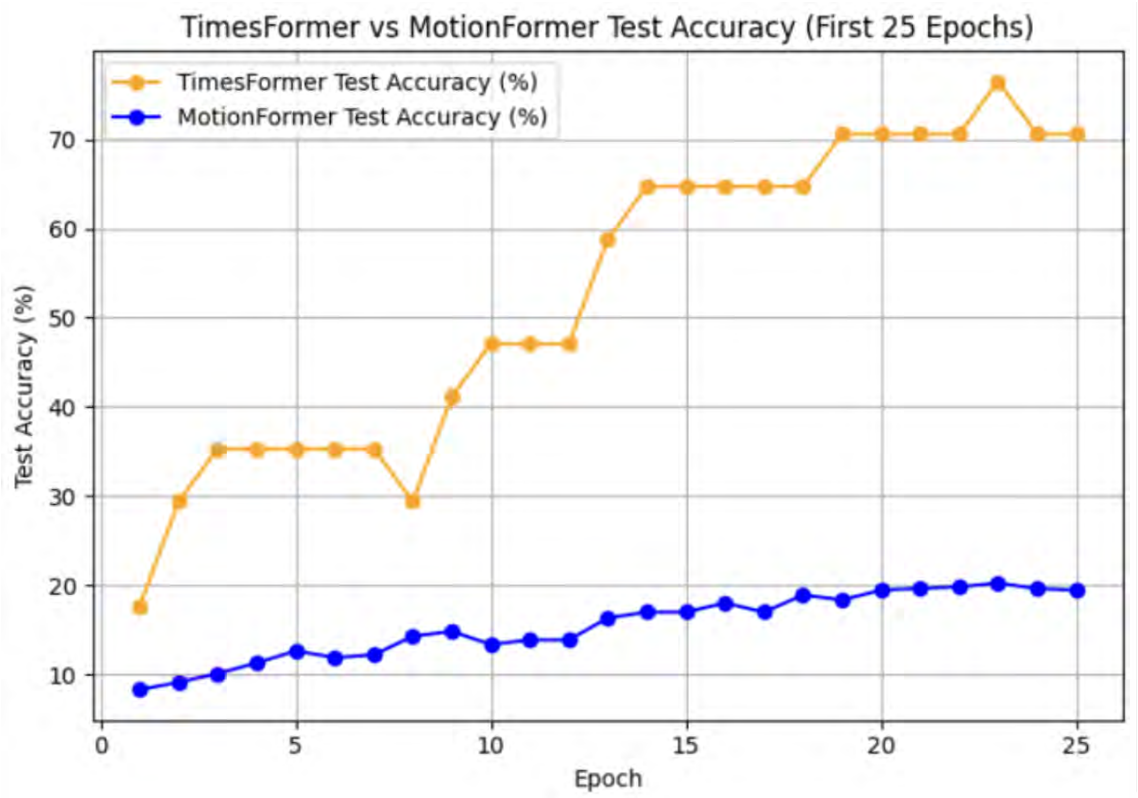


Figure 4.5: *TimeSformer VS MotionFormer Test Accuracy*

### 4.3.4 MotionFormer Implementation:

**Data Pre-processing:**
In both TimeSformer and MotionFormer, appropriate data preprocessing is necessary to make sure that the models are fed with good quality input. The preprocessing steps that were used in the implementation are detailed below:

1. **Frame Extraction:**
   - All the videos in the dataset are fractured into separate frames.
   - These frames are then stored in class-specific folders (e.g., Abuse, Explosion) and this assists the model to match its frames to the respective categories of anomaly.
2. **Resizing and Normalization:**

- Resizing of frames to 224x224 pixels is done to have uniformity among all the inputs.
- The pixel values are brought to either the range of [0, 1] or and [-1, 1] as per the need of the model.

3. **Foreground Masking (SAM2):**
    - Foreground masks are generated using a model named Segment Anything Model 2 (SAM2) and these emphasis areas are object-centric, human-centric and action-relevant. This minimizes noise in the background and better detection of anomalies.
4. **Optical Flow Generation (S-Flow):**
    - Separable Flow (S-Flow) is then used to produce the optical flow maps after the extraction of foreground masks. It assists in capturing the motion dynamics, which can give temporal information of the movement of the objects between neighboring frames.
5. **Dataset Structuring:**
    - The frames will be divided into train/test split wherein 70% of the data will be utilized in training and 30% will be applied in testing.
    - The classes (e.g., Abuse, Explosion) have around 45 videos and the video is cut into frames which have a rate of 5 frames per second.

**Fine Tuning:**
Fine-tuning can be relevant to adjust the pre-trained models (TimeSformer and MotionFormer) to the particular task of anomaly detection. This process is comprised of a number of steps:

1. **Head-Only Fine-Tuning:**
    - The first step involves the training of only the classification head of the model and holding the backbone constant. This assists the model to determine how to categorise the type of anomaly according to the previously learnt features of the backbone.
2. **Partial Fine-Tuning:**
    - During this stage, additional layers of the model are unfrozen and the model is trained to acquire task-specific properties. This assists the model to fit the specifics of the anomaly detection problem.
3. **Full Model Fine-Tuning:**
    - Lastly, each layer of the model is refined with the help of a small learning rate on the backbone and a large learning rate on the head. It allows the model to be fully customized to fit the dataset with the knowledge of the backbone still present.
4. **Optimizer and Regularization:**

- The weights of the model are updated with AdamW optimizer.
- To reduce overfitting and enhance generalization techniques such as regularization such as dropout and label smoothing are used.

5. **Data Augmentation:**

   - Throughout the training process, several augmentations like RandomResizedCrop, horizontal flip, color jitter and Gaussian blur are used to ensure the model is more robust, and there is no overfitting.

### 4.3.5 TimeSformer vs SlowFast

**SlowFast:**
SlowFast model was created to recognize actions and detect anomalies in video data specifically. It proposes a dual-stream architecture, which is a video processing line the video is processed at two temporal resolutions, a slow path and a fast path. This design enables SlowFast to be able to capture the appearance (spatial features) and the motion (temporality dynamics). The slow route is a way of capturing the static nature of the scene whereas the fast route is the way of capturing the dynamics of movement either in the scene or in the rapid change of the scene.

1. **Dual-Stream Architecture:**

   - **Slow Path:** This path operates with a lower temporal resolution (e.g. one frame in 16 frames), and captures features in the appearance (spatially).
   - **Fast Path:** This path processes at a rate which is higher in terms of time (e.g. 1 frame in every 4 frame) and captures motion features (temporal dynamics).

2. **Cross-Modal Attention:** The outputs of both paths are then merged together after being processed by separate encoders with the help of cross-modal attention, allowing the model to learn the spatial and motion feature interactions.

SlowFast is good at learning both motion-based anomalies because it captures the fast path and also because it learns the spatial features because it captures the slow path. The model is adapted to effectively run both high and low-temporal resolutions on the video and hence it is capable of identifying appearance and motion based anomalies.

**SlowFast Model Architecture:**

1. **Dual Paths (Slow and Fast):**

   - The slow path also runs at a reduced frame rate (e.g. 16 frames per second) and derives appearance features.
   - The fast-path uses an increased frame rate (e.g. 4 frames per second) and derives motion features.

2. **3D Convolutional Layers:**

- Each of the two tracks employs 3D convolutions to learn both the motion dynamics and the appearance at rest. 3D convolutions enable the model to learn the motion, as well as the appearance of an object.

3. **Fusion of Features:**

   - The features of both paths are merged after being processed by their respective encoders, using cross-path connections. This combination allows the model to combine space and time data to detect anomalies.

4. **Training Efficiency:**

   - Although SlowFast is more costly to compute, as it has dual-stream, it can support the fine-grained motion capture using the fast path and fine appearance learning using the slow path, making it ideal in motion-related tasks.

### 4.3.6 SlowFast Implementation:

SlowFast was used on our 14-class video anomaly dataset, an input of RGB frames and optical flow (created by S-Flow) of each video. The implementation and work with the dataset at SlowFast was done as follows:

**Data Preprocessing:**

1. **Frame Extraction:** The videos were extracted in single frame with a sampling rate of 5 fps.

2. **Resizing and Normalization:** The frames were resized to 224x224 pixels and normalised to make sure that all inputs had similar sizes.

3. **Optical Flow Generation:** A motion dynamic of capturing motion was made using S-Flow to develop optical flow maps.

4. **Foreground Masking:** SAM2 was used to concentrate the model on object centric features in the frames and eliminate the background noise.

**Fine-Tuning:**

1. SlowFast was fine-tuned using a two-phase strategy:

   - **Phase 1:** Optimize the classification head keeping the backbone fixed to learn quickly the model on our data.
   - **Phase 2:** Freeze the complete model and include all the layers and make it fine-tune to acquire additional specific features associated with the types of anomalies in the dataset.

2. Training and Evaluation:

   - A custom training loop was used to train the model, with gradient accumulation and class-weighted sampling to streamline the training procedure, particularly due to the imbalance in the number of classes in the dataset.

- The data augmentation methods that we used during training were horizontal flips, random crops, and color jitter to enhance generalization.
- The model was tested on test data, with a good performance in the motion-heavy anomaly tasks, including fights and explosions, but it took more training epochs to reach a stable performance in the general anomaly detection tasks.

**Comparative Analysis:**

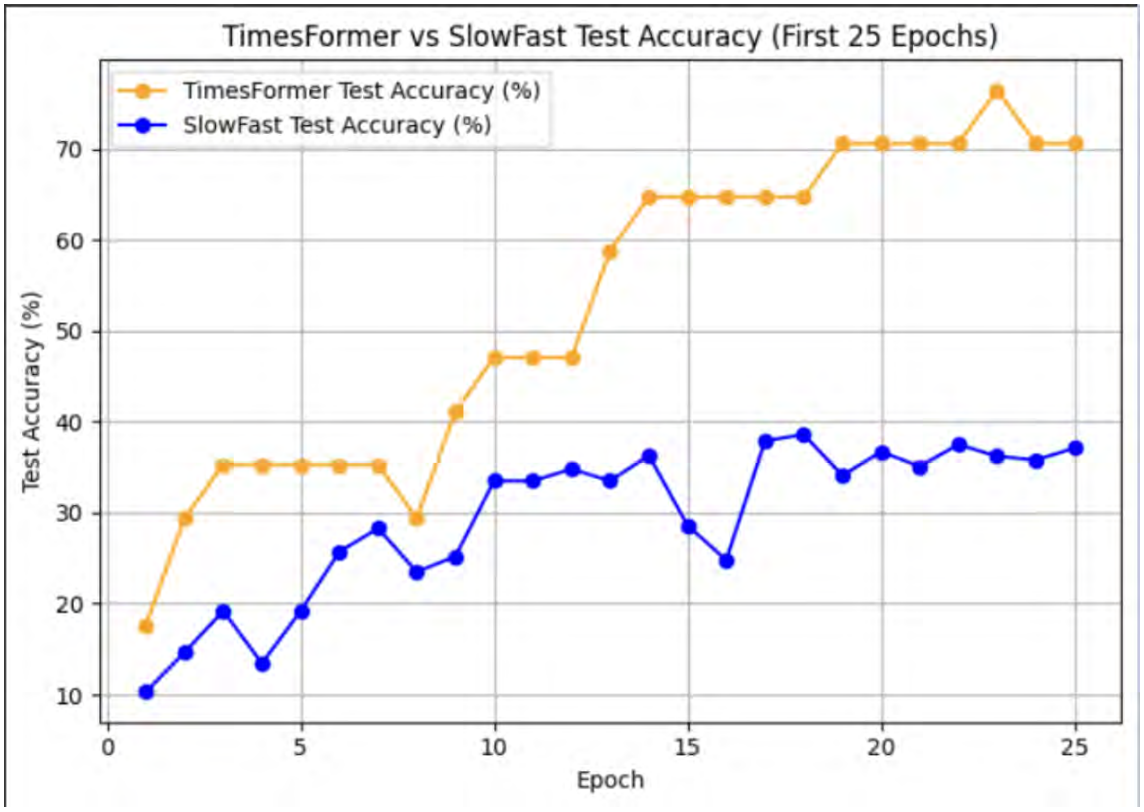


Figure 4.6: *TimeSformer VS SlowFast Test Accuracy*

SlowFast starts near 14%, climbs more gradually into the mid-20s during early epochs, reaches the mid-to-high 30s in the late teens, and ends around 38% at epoch 25.From about epoch 10 onward, Timesformer maintains a 25–35 percentage-point lead, reflecting a stronger and more stable learning trajectory.Both models improve with training, but Timesformer shows sharper gains and a higher ceiling, indicating superior effectiveness for the depicted setup.

### 4.3.7 TimeSformer Model Architecture

TimeSformer is an extension of the Vision Transformer (ViT) configuration, which is developed specifically to process video information. ViT models have been reported to be successful at performing image classification tasks by viewing each image as

a sequence of non-overlapping patches, which are then linearly embedded to a flat sequence and processed using a transformer-based architecture. TimeSformer alters this idea to extract the spatial and temporal dependence among video data.

**Spatial and Temporal Attention Mechanism**

- **Spatial Attention**:
  The model of the traditional ViT is such that the input picture is scraped into patches (e.g. 16x16 pixels) and every patch is flattened into a vector and passed through a transformer model. On the same note, in **TimeSformer**, every frame (image) is considered as an array of patches. Spatial attention system is trained to pay attention to various parts of the frame to obtain spatial patterns. These patches are subjected to the self-attention.

Mathematically, spatial attention in TimeSformer can be described as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{4.2}$$

Where:

The tokens of the image in the spatial domain are

- $Q$ (queries), $K$ (keys), and $V$ (values). The key vectors have a dimension of $d_k$. This enables the model to acquire spatial information such as object boundaries, shapes as well as textures in each frame.

This allows the model to capture **spatial relations** like object boundaries, shapes, and textures within each frame.

- **Temporal Attention**: In the case of videos, TimeSformer poses time attention to learn the relationships among frames. This implies that, besides focusing on spatial features in each frame, TimeSformer will learn the way that frames form a sequence with respect to time. This plays a very significant role in comprehending the movement and dynamics of a scene.

  Temporal attention can be represented as:

$$\text{Temporal Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{4.3}$$

Where:

- $Q$, $K$, and $V$ are tokens of different frames.
- The model is able to record temporal dependencies like motion, object movements or even frame transition by using attention over the time points.

Multi-head attention is employed by TimeSformer and enables the model to acquire various features of the relationships between the spatial and time dimensions.

### 4.3.8 BLIP-2 Model Architecture(Bootstrapping Language-Image Pretraining)

One of the most recent models with multimodality is called BLIP-2 (Bootstrapping Language-Image Pretraining) and is aimed at tasks that require both vision and language. It combines vision transformers (ViTs) with the most recent state-of-the-art vision encoders with language decoders based on transformers and incorporates a new vision-language fusion mechanism that allows learning joint representations more effectively.

Vision-Language Pretraining (VLP) has become an important methodology in the development of machine concepts of multimodal information. BLIP-2 is a leap in visual and textual modalities fusion, and it is able to provide more effective and scalable models to handle image captioning, visual question answering (VQA), and image-text retrieval. BLIP-2 is also a development of the Transformer architecture, which is a popular framework in computer vision and natural language processing, in that it combines both visual information and textual inputs into a single framework.

**BLIP-2 consists of two primary modules:**

1. **Vision Encoder**
2. **Language Decoder**

**Vision Encoder:**
The Vision Transformer (ViT) is used as a Vision Encoder of BLIP-2 to process the visual input. The image is initially split into non-overlapping patches, those are encoded in a sequence of tokens linearly. Transformer encoder is then used to extract contextual embeddings of the image using these tokens.

Mathematically, the process is represented as follows:

$$\mathbf{I = \text{Image Input}}$$

The image is split into $N$ patches, each of size $P$X$P$, and each patch

$$x_i = \text{PatchEmbed}(I) \quad \text{for} \quad i = 1, 2, \ldots, N \tag{4.4}$$

The embedded patches are then fed to the Vision Transformer encoder which does self-attention on the patches, contextualizing them:

$$\mathbf{Z}_{\text{image}} = \text{ViT}_{\text{encoder}}(x_1, x_2, \ldots, x_N) \tag{4.5}$$

Here, $Z_{image}$ is the visual embedding outputs that reflect spatial and semantic image features.

**Language Decoder:**

BLIP-2 Language Decoder is also based on a Transformer decoder, e.g., BART, that takes natural language queries or captions. Textual inputs are tokenized and fed through an embedding layer to produce textual embeddings which are subsequently fed through the decoder.
Let the input text be denoted as

$$t = [t_1, t_2, \ldots, t_M] \tag{4.6}$$

where each $t_i$ is a token in the input. These tokens are incorporated in the following way:

$$t_{\text{emb}} = \text{Embed}(t_1, t_2, \ldots, t_M) \tag{4.7}$$

The Transformer decoder processes the text embeddings and the attention mechanism makes sure that the text is able to pay attention to the visual features:

$$Z_{\text{text}} = \text{Transformer}_{\text{decoder}}(t_{\text{emb}}, Z_{\text{image}}) \tag{4.8}$$

The output $Z_{\text{text}}$ is the contextualizing textual embedding that is filled with the information that is retrieved both through the text and the visual input.
**Vision-Language Fusion**
The essence of the innovation of BLIP-2 is the mechanism of fusion of vision and language. This process allows connection between the image and text embeddings, which is cross-attentional such that text query (or caption) focuses on vision, and the image embeddings are also focused on the text input.
Cross-attention can be formulated as:

$$\text{CrossAttention}(Q_{\text{text}}, K_{\text{image}}, V_{\text{image}}) = \text{softmax}\left(\frac{Q_{\text{text}} K_{\text{image}}^T}{\sqrt{d_k}}\right) V_{\text{image}} \tag{4.9}$$

Where:

- $Q_{\text{text}}$ is the query (textual embedding),
- $K_{\text{image}}$ and $V_{\text{image}}$ are the key and value from the image embeddings, respectively,
- $d_k$ is the dimensionality of the key vectors.

This fusion step makes it possible to align both modalities, thus the model is able to produce more precise captions, solve visual queries, or do other multimodal tasks.

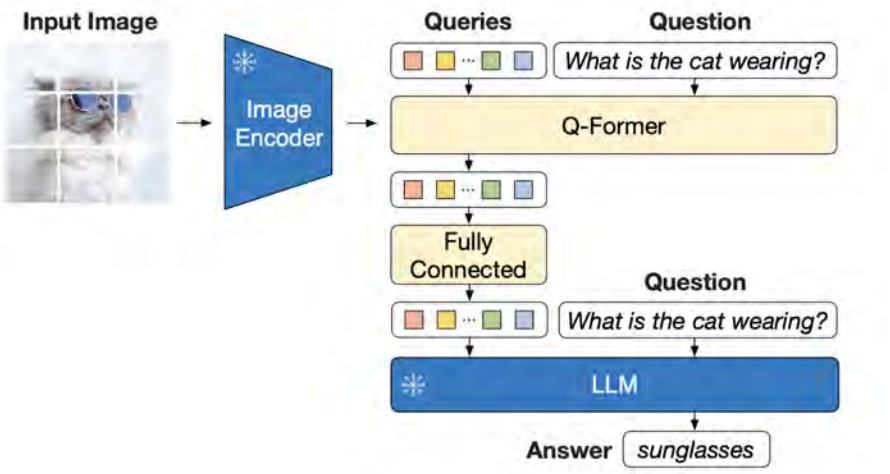


Figure 4.7: BLIP-2 Model Architechture

### 4.3.9 BART Model Architecture

BART (Bidirectional and Auto-Regressive Transformers) is a general and strong sequence-to-sequence model developed to be used in text generation, as well as various NLP applications. It combines the advantages of bidirectional and autoregressive Transformers, combining denoising autoencoder pretraining scheme with the autoregressive decoding algorithm. The present paper goes deeper into the design of BART, describes its bidirectional encoder, autoregressive decoder, and the pretraining strategy that allows it to be effective on a wide variety of language tasks.

The sequence-to-sequence model has grown to be the new standard in natural language processing (NLP) tasks such as machine translation, text summarization and question answering. BART is an architecture proposed by the Facebook AI Research that is based on the Transformer architecture with a modification of employing two directions of encoding and autoregressive decoding. The corruption of text during its pretraining enables it to be highly effective in both generation and understanding tasks since it can combine denoising and autoregressive methods to achieve a deep understanding of text and generate coherent and contextually relevant outputs at the same time.

BART is a bidirectional encoder (equivalent of BERT) and an autoregressive decoder (equivalent of GPT). Such a differentiation allows BART to be very flexible in a range of NLP tasks, including summarization, translation, and question answering.

**Encoder:**

The BART encoder is powered by Transformer based on bidirectional training, whereby the encoder trains a model that learns to create contextual representations of the input tokens in the sequence by attending to other tokens. The encoder receives an input sequence and generates a contextualized embedding of each token of the sequence.

Mathematically, the input sequence $X = [x_1, x_2, ..., x_n]$ is passed through the bidirectional Transformer encoder:

$$Z_{\text{enc}} = \text{Transformer}_{\text{encoder}}(X) \tag{4.10}$$

Where $Z_{\text{enc}}$ is the contextualization of the input sequence $X$. The working of this encoder takes both the right and left context of a token into consideration and is powerful to comprehend the relationship between several other sections of the input.

**Decoder:**

Autoregressive is the decoder of BART, however. It produces a token sequentially depending on the already produced tokens and the encoded input of the encoder. The decoder operates by listening to the tokens produced so far and the encoder output in the form of the coded sequence using cross-attention.

Let the input tokens for the decoder at time $t$ be $y_1, y_2, ..., y_t$, where $y_i$ represents the $i$-th token of the output sequence. Each token is generated in an autoregressive manner, depending on the past tokens and the encoder output.

$$y_t = \text{Decoder}_{\text{autoregressive}}(y_1, y_2, \ldots, y_{t-1}, Z_{\text{enc}}) \tag{4.11}$$

$y_t$ is the production token at time $t$ and $Z_{\text{enc}}$ is the history of context-sensitive encoder embeddings. The autoregressive characteristic of the decoder allows the decoder to produce sequences token by token and is therefore applicable in tasks like text generation and text summarization.

**Denoising Autoencoder Pretraining:**

The training process of BART consists of a denoising autoencoding pretraining using text as input where noise functions work under a diverse set of noise functions and the model is trained to restore the original text. The goal of this pretraining is to make the model more robust to downstream tasks by enhancing the sensitivity to a number of input perturbations.

This corruption process has a number of low-level noising functions which include:

- **Token Masking:** This method is used to randomly mask a few tokens and then the model is trained to predict the tokens.
- **Token Deletion:** Simply deleting some tokens in the sequence and training the model to provide the missing values.
- **Sentence Permutation:** This is used to randomize sentence sequence in the input in order to encourage the model to learn sentence-level contextual dependencies.

It is trained in such a way that original sequence $X_0$ can be reconstructed using corrupted input $X_c$ with the help of reducing the loss:

$$\mathcal{L}_{\text{denoising}} = -\sum_{i=1}^{N} \log P(x_i | X_c) \quad \text{where} \quad X_c \text{ is the corrupted input} \tag{4.12}$$

Where the loss is calculated by estimating the original tokens of the noisy input. This pretraining exercise teaches the model to learn and generate a large range of

corruptions of text.

**Combined Encoder-Decoder Architecture:**

Both the encoder and the decoder are Transformer-based. The encoder is used to process the input sequence in both directions to obtain context and the decoder is used to produce the output sequence in an autoregressive manner. The whole model is trained in pretraining and encoder decoder architecture is fine-tuned to be specific to certain applications during fine-tuning.
The architecture of encoder-decoder can be formulated as:

$$Z_{\text{final}} = \text{Decoder}(\text{Encoder}(X)) \tag{4.13}$$

Where $X$ is the input text, $Z_{\text{final}}$ is the final output generated by the model.

**Pretraining and Fine-Tuning:**

BART is trained by a denoising objective, although it can be also fine-tuned to the tasks of interest. The fine-tuning would usually entail:

- **Fine-tuning for Text Generation:** Fine-tuning BART to summarize a long sequence of inputs has been shown to be useful on text summarization problems.
- **Fine-tuning for Text Classification:** In order to overcome such issues, as sentiment analysis, one can fine-tune BART to categorize text based on the context.
- **Fine-tuning for Translation:** In the case of machine translation, BART is fined tuned to translate language to language.

  Fine-tuning involves in the reduction of a task-related loss (e.g. cross-entropy loss in classification, sequence-level loss in generation), through which BART may be tailored to different NLP tasks.

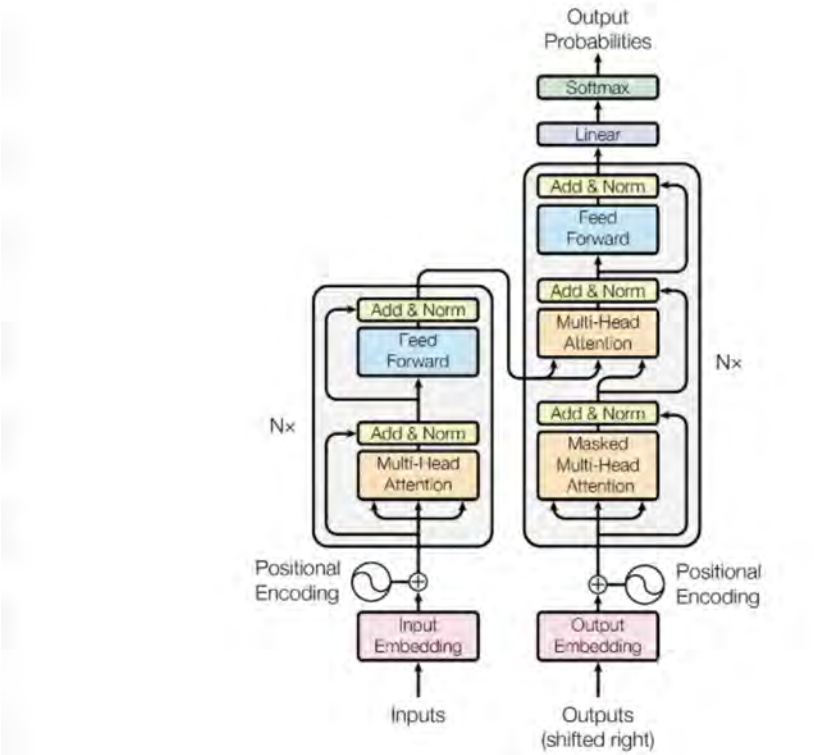


Figure 4.8: BART Model Architechture

### 4.3.10 InstructBLIP Model Architecture

InstructBART is an extension of the BART (Bidirectional and Auto-Regressive Transformers) model, which is used to solve various tasks following the instructions of natural language. It combines the advantages of both bidirectional context encoding and autoregressive sequence generation whilst optimizing on instruction-based fine-tuning. This paper will give an elaborate discussion on the architecture of InstructBART, its pretraining process and mathematical underpinnings that enable it to excel in instruction-following tasks including task-based dialogues, summarization, and multi-step reasoning.
Instruction-following is an important element of Natural Language Processing (NLP) during which the model can be trained to comprehend and implement certain tasks following natural language instructions. InstructBART is a modification of the BART architecture that is fine-tuned on a set of instruction-based tasks. In this way it is optimized to produce text that is as similar as possible to the query instruction, allowing useful task-specific applications e.g. question answering and dialogue systems, as well as multi-step reasoning tasks.

The model structure is based on the BART architecture including task-specific pretraining, instruction-based fine-tuning, which allows the model to comprehend and produce responses to natural language instructions better. This paper entails a detailed discussion of InstructBART, its constituents and its benefits in the instruction-based learning.

**Model Architecture:**

InstructBART is based on the same structure as BART, whose two fundamental parts are:

- **Bidirectional Encoder** (based on Transformer architecture)
- **Autoregressive Decoder** (also based on Transformer)

Nevertheless, InstructBART presents a *task-specific pretraining* phase, which includes a task of instruction-response pairs, and thus proves to be more skillful at instruction following than the original BART model. The encoder of InstructBART is also a **Bidirectional Transformer** architecture like BERT. The encoder is tasked with taking the input instruction (with other related context) and producing context-specific representations of the input sequence.
Given an input $I = [i_1, i_2, \ldots, i_n]$ where each $i_i$ is the token the input sequence is then coded by the bidirectional Transformer encoder to the actual value:

$$Z_{\text{enc}} = \text{Transformer}_{\text{encoder}}(I) \tag{4.14}$$

Here, $Z_{\text{enc}}$ is the contextualized implementation of the sequence of instructions. The bi-directional encoding enables the model to be fully aware of the context of each token in the instruction and therefore is useful in tasks that need the awareness of the whole input context.
**Decoder:**
InstructBART has an autoregressive decoder, like BART, which is intended to produce one token at a time. It gets input the encoded message of the encoder and the tokens that have already been created as a result of the autoregressive generation process.
Each result of the encoder output $Z_{\text{enc}}$ and a series of tokens of the autoregressive process produces the successive token of the sequence, conditional upon the generated preceding tokens and the encoded instruction. The expression of the decoding can mathematically be expressed as:

$$y_t = \text{Decoder}_{\text{autoregressive}}(y_1, y_2, \ldots, y_{t-1}, Z_{\text{enc}}) \tag{4.15}$$

Where $y_t$ is the token produced in time step $t$ and $Z_{\text{enc}}$ is a list of the context-sensitive embeddings (produced by the encoder). The autoregressive property of the decoder enables it to produce the output of the instruction step by step which is necessary in multi-step or complex tasks.

**Task-Specific Pretraining:**
The main aspect of InstructBART is the task-based pre training stage. At pretraining, InstructBART is shown a set of instruction-response pairs, which enables it to learn to produce outputs in response to natural language instructions in an effective manner.
The pretraining task of InstructBART implies corruption of sequences of instruction-response pairs and the training of the model to restate the original text. This has been done by using a *denoising autoencoding* method, which involves masking or substituting some sections of the input sequence and the model is trained to infer the missing sections. The pretraining purpose is to reduce the loss as given below:

$$\mathcal{L}_{\text{denoising}} = -\sum_{i=1}^{N} \log P(y_i|X_c) \quad \text{where} \quad X_c \text{ is the corrupted input} \tag{4.16}$$

This training method causes the model to acquire the association between instructions and their related responses, and this enhances the capability of the model to comprehend and perform a multiplicity of different tasks according to instructions.

**Cross-Attention Mechanism:**

Cross-attention between the encoder and decoder is a technique of InstructBART to bridge successfully the gap between the instruction and its resultant output. The model can generate response (output) and match it with instruction (input) in a way that serves as a representation of the relationships between the two modalities through the cross-attention.

Cross-attention works by focusing on the message that the encoder produces as part of the instruction sequence and improving the process of token generation by this input. The attention is mathematically determined as:

$$\text{CrossAttention}(Q_{\text{text}}, K_{\text{enc}}, V_{\text{enc}}) = \text{softmax}\left(\frac{Q_{\text{text}} K_{\text{enc}}^T}{\sqrt{d_k}}\right) V_{\text{enc}} \tag{4.17}$$

Where:

- $Q_{\text{text}}$ is the query (generated tokens in the decoder).
- $K_{\text{enc}}$ and $V_{\text{enc}}$ are encoder (instruction embeddings) key and value respectively.
- $d_k$ is the number of dimensions of the key vectors.

This process makes sure that the decoder focuses on the most important aspects of the instruction during producing the output, which allows the model to execute multi-step and intricate instructions.

**Applications**

InstructBART has shown remarkable performance in comparison with the conventional generative models in a broad spectrum of applications, such as:

- **Text Summarization:** InstructBART is able to summarize long text (based on high-level instructions, e.g., "Summarize this article in three sentences").
- **Task-based Dialogue Systems:** It is capable of taking instructions to do things in a dialogue system, like answering particular questions or giving explanations.
- **Multi-step Reasoning:** InstructBART is strong in the tasks that involve the stepwise reasoning, e.g. puzzle solving or following instructions.
- **Code Generation:** It allows refining the model to produce snippets of code with reference to natural language requirements.

**Mathematical Formulations for Task-Specific Learning**

The denoising and sequence generation is combined with fine-tuning pretraining and task-specific pretraining of InstructBART. The mathematical concepts that will be important during the training of the model are:

- **Attention Mechanism (Self-Attention and Cross-Attention):**

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{4.18}$$

- **Denoising Pretraining Objective:**

$$\mathcal{L}_{\text{denoising}} = -\sum_{i=1}^{N} \log P(y_i | X_c) \tag{4.19}$$

Where $X_c$ is corrupted as input sequence, and the model to predict original tokens $y_i$.

- **Autoregressive Sequence Generation:**

$$y_t = \text{Decoder}_{\text{autoregressive}}(y_1, y_2, \ldots, y_{t-1}, Z_{\text{enc}}) \tag{4.20}$$

Where $y_t$ is the token generated at time step $t$, and the model generates tokens one at a time in an autoregressive manner.

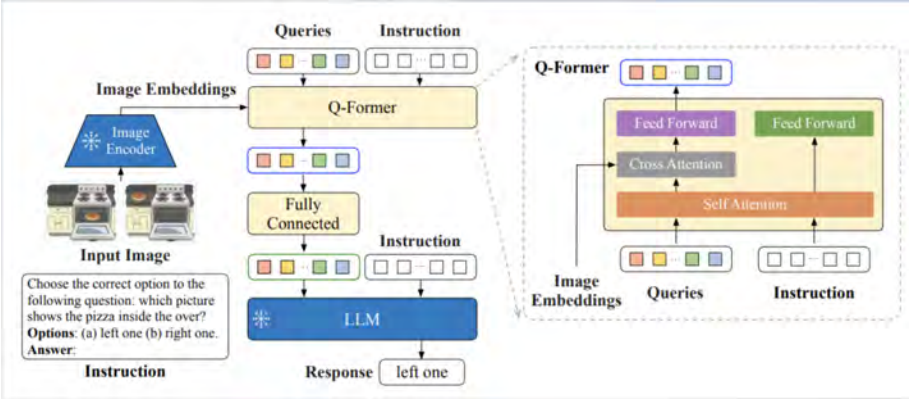


Figure 4.9: InstructBLIP Model Architechture

### 4.3.11 OpenCLIP Model Architecture

OpenCLIP is a vision-language pretraining model that is based on producing shared multimodal embeddings between images and text. It is based on the CLIP (Contrastive Language-Image Pretraining) model, but with a more scalable implementation being open-source. OpenCLIP has a dual-stream structure with both vision and language inputs being projected into a common vector space through large-scale contrastive learning.OpenCLIP is more flexible and scalable than earlier implementations, and it has access to pre-trained models that can be fine-tuned to a variety of downstream uses.

OpenCLIP architecture has been constructed out of two key components:

- **Vision Encoder:** This is a neural network that takes in image responses and generates some visual features.

- **Text Encoder:** This is a neural network that takes textual as inputs and outputs the semantic embeddings.

The model is trained through contrastive learning and the embeddings of the images and the corresponding textual descriptions lie in a common vector space.
**Model Architecture** There are 2 central components of OpenCLIP:

- **Vision Encoder:** It is usually built on a Vision transporter (ViT) or ResNet architecture.
- **Text Encoder:** On a Transformer architecture.

Both encoders are simultaneously trained with contrastive learning to project both of the representations in the same space. The model is made very flexible and can be used to both pretrain vision-language tasks at large scale as well as fine-tuning.
**Vision Encoder**
The Vision Encoder in OpenCLIP does the processing of visual data i.e. images and converting them into fixed length vectors. OpenCLIP can use various vision encoder architecture based on the size and performance of the task, such as Vision Transformers (ViT), or ResNet.

- **Vision Transformer (ViT)** is typically applied to its capability to simulate long-range correlations and intricate reactions on picture patches.

Given an input image $I$, the image is first divided into $N$ non-overlapping patches of size $P \times P$ which are then embedded in sequence linearly into a set of tokens. These tokens are run in a sequence of Transformer layers to produce the representation of an image $Z_{\text{image}}$.
Mathematically; the vision encoding process can be defined as:

$$x_i = \text{PatchEmbed}(I) \quad \text{for} \quad i = 1, 2, \ldots, N \tag{4.21}$$

$$Z_{\text{image}} = \text{ViT}_{\text{encoder}}(x_1, x_2, \ldots, x_N) \tag{4.22}$$

Where $x_i$ are the image patches that are embedded into tokens, and $Z_{\text{image}}$ the output of the vision transformer that is a vector representation of the image.
**Text Encoder:**
The Text Encoder is used to process textual data (e.g. captions or queries) to create specific text embeddings. The text encoder in OpenCLIP is usually a Transformer-based model, which is comparable to other models such as BERT and GPT, and is capable of complex sentence structure and word dependence.
Given an input sentence $T = [t_1, t_2, \ldots, t_M]$, the sentence is tokenized, and a token is assigned to each token. These token embeddings are then sent through a number of Transformer layers in order to produce the text representation $Z_{\text{text}}$.

$$t_{\text{emb}} = \text{Embed}(t_1, t_2, \ldots, t_M) \tag{4.23}$$

$$Z_{\text{text}} = \text{Transformer}_{\text{encoder}}(t_{\text{emb}}) \tag{4.24}$$

Token embeddings are where tokens are the token embeddings, and the final output representation of the $\text{t}_{\text{emb}}$ is $Z_{\text{text}}$.

**Cross-Modal Contrastive Learning:**
The core of the OpenCLIP architecture is the **contrastive learning objective** objective which promotes the match of the image and text representations in the common embedding space. In the model, we learn to maximize the similarity between pairs of image texts (positive pairs) and minimise the similarity between non-paired pairs (negative pairs). This makes OpenCLIP produce embeddings in which images and description are closer to each other in the shared space than are irrelevant pairs of images and text.
The contrastive loss is defined as:

$$\mathcal{L}_{\text{contrastive}} = -\sum_{i=1}^{N} \log \frac{\exp(\text{sim}(Z^i_{\text{image}}, Z^i_{\text{text}})/\tau)}{\sum_{j=1}^{M} \exp(\text{sim}(Z^i_{\text{image}}, Z^j_{\text{text}})/\tau)} \tag{4.25}$$

Where:

- $\text{sim}(Z_{\text{image}}, Z_{\text{text}})$ is the cosine similarity between the image and text embeddings,
- $\tau$ is a temperature scaling factor,
- $Z^i_{\text{image}}$ and $Z^i_{\text{text}}$ are the embeddings of the $i$-th image and text pair,
- $M$ is the total number of negative samples.

The contrastive loss ensures that the embeddings of similar pairs of images and texts are closer to each other, whereas non-similar pairs are separated.
**Cross-Modal Retrieval:**
The OpenCLIP is mainly used in cross-modal retrieval when the model has to retrieve the relevant images using a text query or vice versa. In inference, OpenCLIP takes the image encoder and the text encoder to create embeddings on the images and textual queries. After this, these embeddings are used to compare them with cosine similarity to the purpose of identifying the most appropriate image or text.

The similarity between an image $I$ and text query $Q$ is calculated as:

$$\text{similarity}(I, Q) = \frac{Z_{\text{image}} \cdot Z_{\text{text}}}{\|Z_{\text{image}}\| \, \|Z_{\text{text}}\|} \tag{4.26}$$

Where $\cdot$ denotes the dot product, and $\|\cdot\|$ represents the L2 norm. The model recalls the image or text which is the closest one to the input query.

OpenCLIP is based on the principle of contrastive learning and the most important mathematical process of the model is the calculation of the cosine similarity between embedding image and text representations:

$$\text{sim}(Z_{\text{image}}, Z_{\text{text}}) = \frac{Z_{\text{image}} \cdot Z_{\text{text}}}{\|Z_{\text{image}}\| \, \|Z_{\text{text}}\|} \tag{4.27}$$

The contrastive loss is an objective that compels the model to bring similar image-text pairs nearer in the common space and distinguish dissimilar ones:

$$\mathcal{L}_{\text{contrastive}} = -\sum_{i=1}^{N} \log \frac{\exp\big(\text{sim}(Z^i_{\text{image}}, Z^i_{\text{text}})/\tau\big)}{\sum_{j=1}^{M} \exp\big(\text{sim}(Z^i_{\text{image}}, Z^j_{\text{text}})/\tau\big)} \tag{4.28}$$

Where $\tau$ is the temperature scaling term that is used to regulate the sharpness of the similarity distribution.

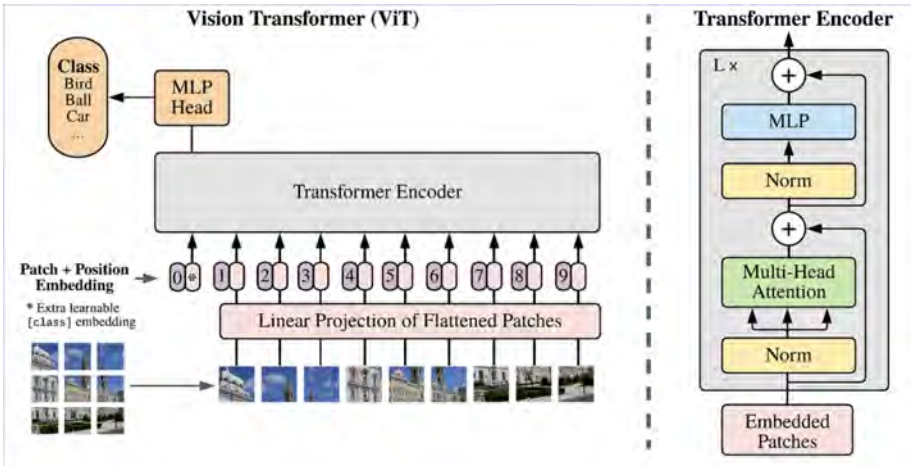


Figure 4.10: OpenClip Model Architechture

## 4.4 Model Pipeline

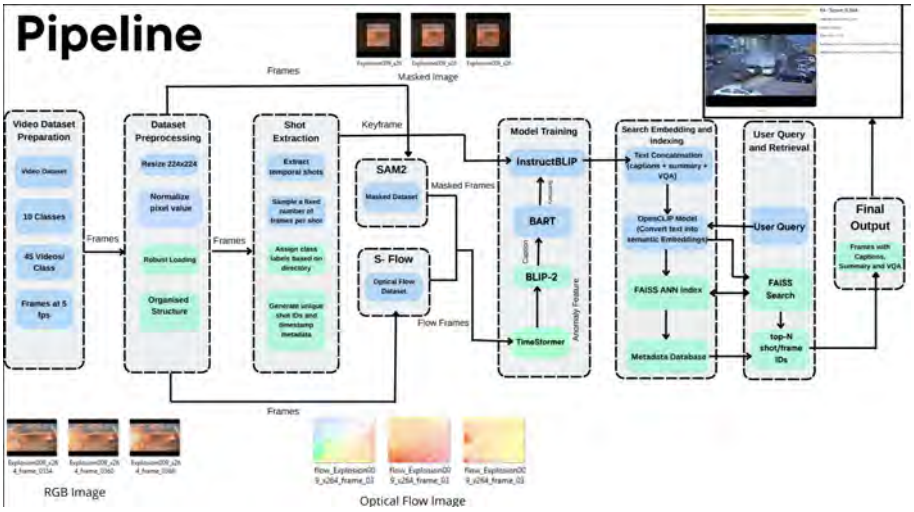


Figure 4.11: Whole Work-Flow Pipeline

**TimeSformer** TimeSformer is employed as the base model of **anomaly classification**. The **RGB frames** (spatial features) and **optical flow frames** (motion dynamics) are first pre-processed and then combined and inputted to TimeSformer to be classified. This facilitates the model to use **temporal patterns** (appearance) and **temporal patterns** (movement) to identify anomalies (violence, accidents or abnormal behavior) within space and time.

The pipeline uses both **spatial and temporal attention** to classify every video **shot** (3 seconds of video) according to the features obtained in the fused frames. The TimeSformer model provides the class probabilities of each shot which is the probability of the shot to fall into one of the classes of anomalies that have been decided on.

**Detailed Steps:**

1. **Input preparation:** The raw video data is separated into classes, each one of which is a variant of an anomaly (e.g.,**Abuse**, **Explosion**, **RoadAccident**). The classes folders have about 45 video files in each and the video files are transformed into frames that have been extracted at **5 frames per second (fps)** each. These frames constitute the input of the pre-processing and the training pipeline.

2. **Dataset Organization (Train/Test Split):** To make sure that the model can be generalized, the video dataset is divided into **training** and **test** ones. The data will be organized in the following way:

3. **Data Preprocessing:** Pre-processing of the data makes the entry data to be in the proper form that the models can process and train.

    - **3.1 Resize and Normalize Frames:** The frames have been made equal size (224x224 pixels) and scaled so that the pixel values are within the same range (between 0 and 1). This is imperative in feeding the frames into time transformers such as TimeSformer.

      Mathematically:

      $$\mathbf{X}_{\text{norm}} = \frac{\mathbf{X} - \mu}{\sigma} \tag{4.29}$$

      where:

        - $\mathbf{X}$ is the unprocessed values of the frame pixels.
        - $\mu$ is the average value of the pixel values of the dataset.
        - $\sigma$ is the standard deviation between the pixel values of the dataset.

    - **3.2 Temporal Frame Sampling:** Sampling and sampling frames are done deterministically (to ensure consistency) and randomly (to ensure data augmentation), to ensure a balanced temporal sampling.

    - **3.3 SAM2 Mask Dataset Generation:** The model produces object-centric, human-centric, and action-centric masks with the help of the Segment Anything Model 2 (SAM2) and enables the model to prioritize the parts of the frame that are pertinent to the anomaly being detected (e.g., a person or an object in an accident).

    - **3.4 S-Flow Optical Flow Dataset:** Separable Flow (S-Flow) is an optical flow calculation used between successive frames, which is used to estimate the dynamics of motion of the scene. Optical flow is necessary

to recognize movements in the video that can indicate an anomaly (e.g., a car crashing, a person running).

4. **Feature Fusion:** After the preprocessing of the data, the optical flow frames and the RGB frames are merged. The combined representation enables the model to acquire the spatial (appearance) and the temporal (motion) features that are essential in identifying the anomalies.

   Mathematically, the fused feature input is:

$$\mathbf{X_{fused}} = [\mathbf{X_{RGB}}, \mathbf{X_{flow}}] \tag{4.30}$$

   Where $X_{RGB}$ is a spatial feature (of RGB frames), and $X_flow$ is a temporal feature (of optical flow).

5. **TimeSformer Model Implementation:**

   i. **Backbone and initialization:** Based on TimeSformer video transformer, it is possible to use public checkpoints (e.g., Kinetics-400 fine-tuned or ImageNet-initialized variants) and then fine-tune on the anomaly dataset to adapt spatiotemporal attention to CCTV domains.

   ii. **Foreground masking:** Use SAM 2 to create consistent and promptable object and motion masks in each frame such that the model concentrates on incident relevant areas whilst ignoring background noise during training clips.

   iii. **Two-stream inputs:** Train two stream (RGB frames and motion stream (optical flow)) with a two-stream fusion approach in order to use temporal dynamics to supplement appearance features in detecting anomalies.

   iv. **Fine-tuning procedure:** If not, do using standard practices in transformer training:batch, fixed-length clip sampling, AdamW-based optimization, regularization Image emerging as typically in TimeSformer training Simple-minded tuning Fine-tune common practices in transformer training.

   v. **Validation and selection:** We pick up checkpoints according to held-out performance, which are a set of monitoring clip-level classification metrics that are conditioned on the description-conditioned pipeline that is prepared with the thesis.

**5.1 Predictions:**

   i. **New footage classification:** Turn an incident description into a text prompt of the description-conditioned pathway and run TimeSformer inference to get class probabilities and anomaly scores of new CCTV clips.

ii. **Incident matching:** Ranking of the clips or parts according to their relevance to the description is done using the predicted classes probabilities and description matching scores to highlight the footage which best suits the described incident.

iii. **Confidence levels:** In case of lower confidence of the match, re-process the clip refining SAM 2 masks and/or enlarge clip length or spatial resolution to enhance features extraction and re-assess it.

### 5.2 Outputs:

i. Clip-level logits and statistics on the anomaly label space that can be used to threshold, rank or downstream decision rules in the CCTV case.

ii. The anomaly scores of the segments over the timeline to localize intervals of interest concerning an incident to be reviewed and retrieved.

iii. TimeSformer spatiotemporal attention maps that can be interpreted as qualitative explainability of what parts of the image and at what time slot the decision was made.

iv. Relevance scores based on description which corresponds to the description that the prompt conditioned the use of to determine the extent to which the clip is similar to the incident description that the user provided.

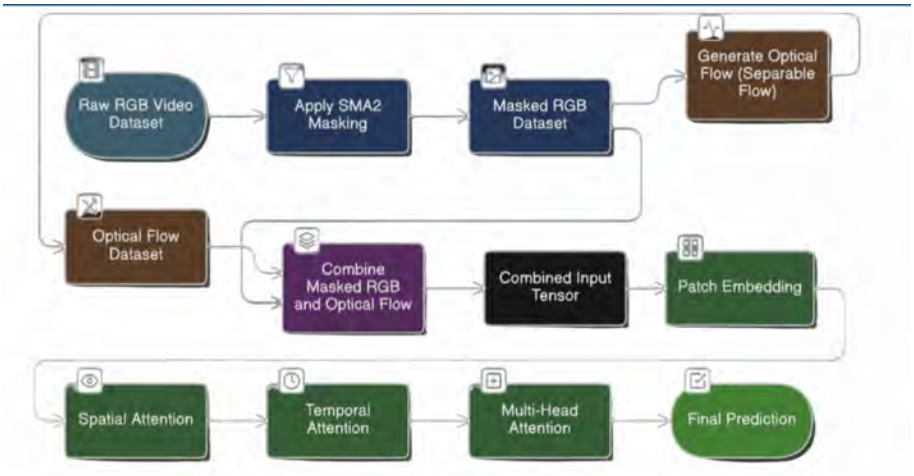


Figure 4.12: TimesFormer WorkFlow

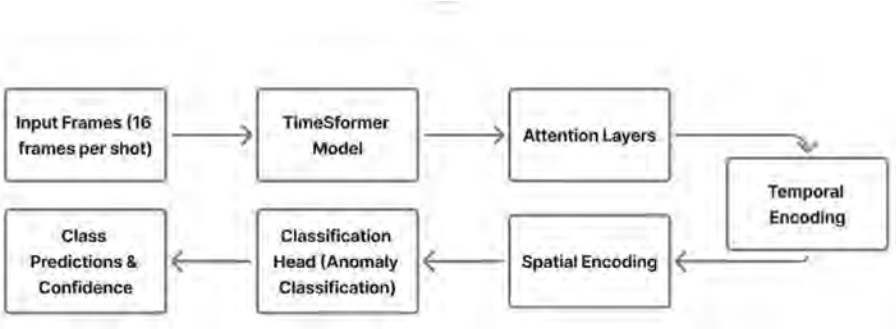


Figure 4.13: Anomaly Classification With TimesFormer

6. **Shot Extraction:**
   Video frames in this step are separated into smaller temporal units known as shots, which show different portions of the video. Every shot passes through the given processes:

   - **Sliding Window**: The video is examined with the help of a sliding window algorithm in which a 3 seconds sliding window is applied to the successive frames with a stride of 1.5 seconds. This overlapping stride makes sure that the shots in line will have a bit of context of the last part, and the transition between the shots will be easier.
   - **Sample 16 Frames per Shot**: One shot has 16 frames sampled to obtain a representative sample of the visual image. These frames are a compressed version of the shot containing the key information of the image and lessening the computational burden of the inference.
   - **Generate Shot Metadata & Labels**: Metadata (e.g. timestamps, shots length) and labels (e.g. type or category of anomaly) are produced on a shot by shot basis. This is essential in planning the shots and give a context in classification tasks.
   - **Save Shot Information (shots.jsonl)**: Metadata, labels and shot information generated are saved in a `shots.jsonl` file. The file offers a hierarchical format so that it can be easily accessed and analyzed or trained on in the future.
   - **Keyframe Selection (Middle Frame)**: Among the sampled frames, the mid frame is selected as the key frame. This frame is defined as the most representative one of the content of the shot, and the analysis, summarization, and classification take place.

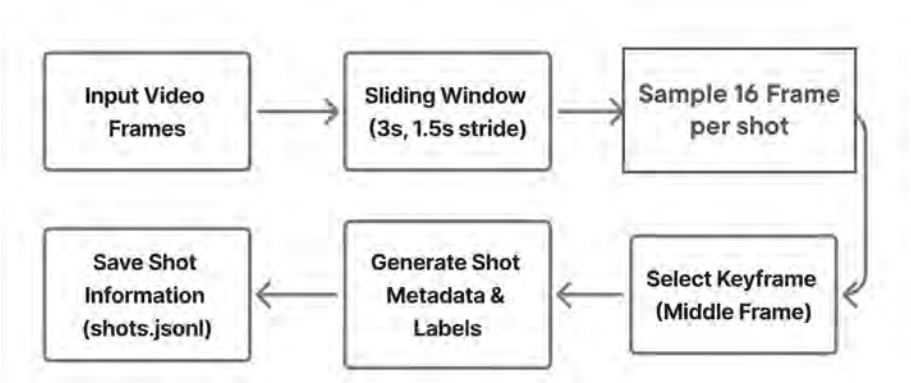


Figure 4.14: Shot Extraction Process Overview

7. **TimeSformer Anomaly Classification:**
   The TimeSformer is applied to every video shot with regards to the selected sequence of 16 frames to decide whether the sequence belongs to one of the 10 categories of anomalies. The model involves the calculation of features at the frame level and in subsequent frames as the time progresses through the implementation of spatial and temporal attention.

   It produces a probability distribution on the 10 classes of anomaly, obtained by the use of the softmax function on the raw model predictions (logits). These logits are transformed into probabilities by the softmax function with the largest value denoting the shot prediction. The model also gives a score of confidence along with the predicted class, this is the probability that the shot belongs to the predicted class.
   In mathematical terms, this process is expressed as:

$$\hat{y}_{\text{shot}} = \text{Softmax}\big(f(\mathbf{X}_{\text{shot}})\big) \tag{4.31}$$

   In which $\hat{y}_{\text{shot}}$ predicts the distribution of the probability of the shot, and f $f(X_{\text{shot}})$ is TimeSformer output logits.

8. **Scene Captioning with BLIP-2:** Image captioning is done using BLIP-2, in which the model produces descriptive captions of every keyframe. These captions give semantic meaning to the anomalies that are detected in the shot. Let $\mathbf{X}_{\text{keyframe}}$ be the input to BLIP-2:

$$\mathbf{C}_{\text{scene}} = \text{BLIP-2}(\mathbf{X}_{\text{keyframe}}) \tag{4.32}$$

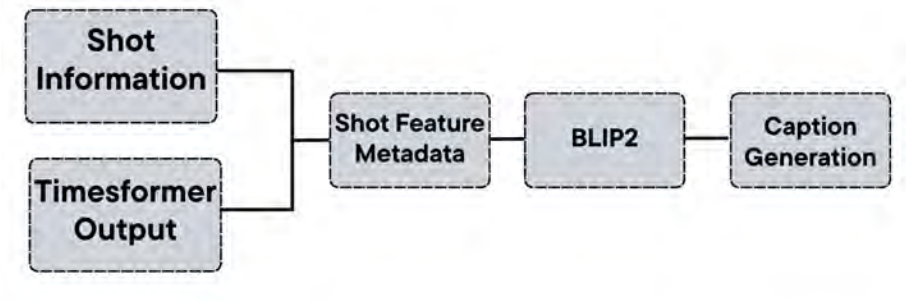


Figure 4.15: Captioning with BLIP-2

9. **Caption Summarization with BART:** After the generation of captions by BLIP-2, it is summarized with BART. BART summarizes several captions into one, coherent sentence, leaving only the necessary information pertaining to the anomalies.

$$\mathbf{S}_{\text{summary}} = \text{BART}(\mathbf{C}_{\text{captions}}) \tag{4.33}$$

BART will put the captions together, and create a summary of each shot, which will provide context to the anomaly.

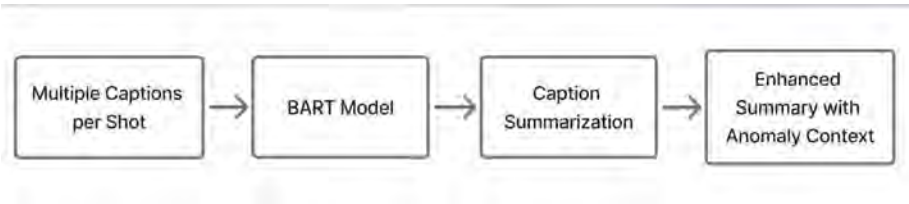


Figure 4.16: Caption Summarization with BART

10. **VQA Enhancement with InstructBLIP:** The creation of detailed answers to specific questions regarding each video keyframe is done with the help of InstructBLIP. These questions are aimed at giving more details on the anomalies in the shot. For each keyframe $\mathbf{X}_{\text{keyframe}}$, InstructBLIP answers five predefined questions:

    - **What is the nature of anomaly occurring?**
    - **Are there human beings there and what are they doing?**
    - **Is it violent, criminal or hazardous?**
    - **Which items/cars are involved?**
    - **Is this a normal or an abnormal situation?**

    The responses are saved along with the metadata of the shot to be utilized later in retrieval and classification.

11. **Embedding and Indexing with OpenCLIP and FAISS:**

### 11.1 OpenCLIP: Embedding Generation:

OpenCLIP is based on the creation of multimodal embeddings, which is a visual representation created with the keyframe and the textual representation created with the captions and VQA answers.
Let:

- $\mathbf{X}_{\mathbf{keyframe}}$ be the image feature.
- $\mathbf{C}_{\mathbf{captions}}$ be the textual caption.

The final fused embedding is given by:

$$\mathbf{E}_{\text{fused}} = \frac{\mathbf{E}_{\text{image}} + \mathbf{E}_{\text{text}}}{2} \tag{4.34}$$

Where, $\mathbf{E}_{\text{image}}$ and $\mathbf{E}_{\text{text}}$ are respectively the image and text embeddings.

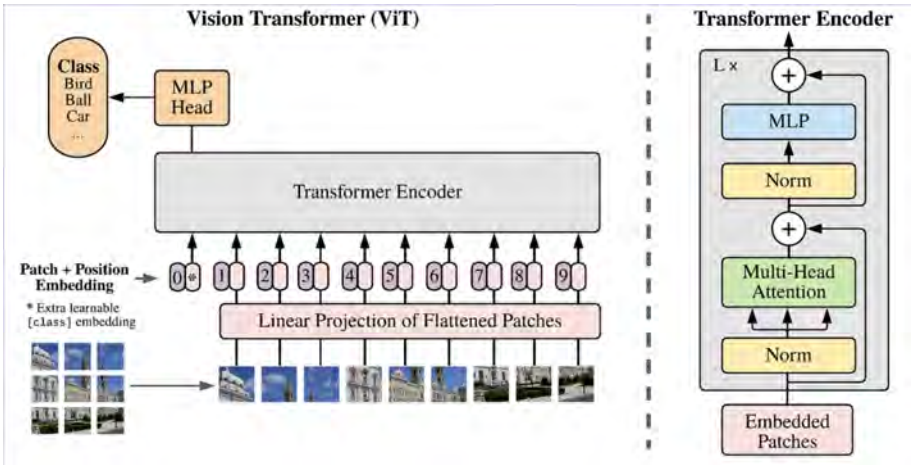


Figure 4.17: OpenCLIP Embedding Generation

### 11.2 FAISS: Efficient Retrieval

These embeddings are indexed by FAISS to be used to search similarity. On a query being supplied, FAISS conducts a similarity search using either cosine similarity or Euclidean distance:

$$\text{Similarity}(q, v_i) = \cos(q, v_i) = \frac{q \cdot v_i}{\|q\| \, \|v_i\|} \tag{4.35}$$

where $q$ is the query vector, and $v_i$ are the indexed video shot embeddings.

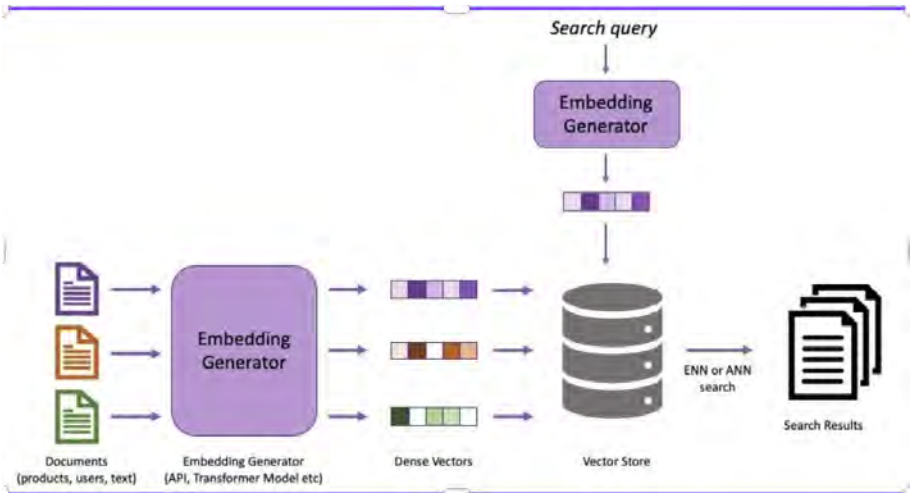


Figure 4.18: FAISS Architecture

12. **Dynamic Query Retrieval:** OpenCLIP embeds the query when the user makes a query, and FAISS makes a search of the most similar video shots. The ranking of results according to the degree of similarity is presented, as well as the display of the relevant video shots, i.e., they are the keyframes, captions, VQA insights, and the predictions of anomalies.

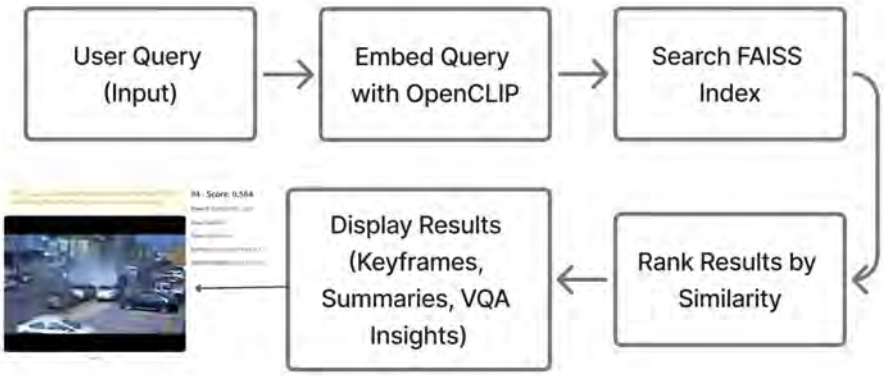


Figure 4.19: Dynamic Query Retrieval

# Chapter 5

# Result and Discussion

The TimeSformer model underwent the operation analysis following the implementation to determine its ability to be used in video surveillance tasks. The received outcomes prove the efficiency of the model and propose the potential ways of improvement.

## 5.1 Performance Metrics Description

The following are the key performance metrics obtained:

- **Accuracy**:
  - TimesFormer was the most accurate of the ones at 0.65, then 0.65 is followed by MotionFormer. SlowFast model had the poorest accuracy of 0.29 and Uniformer had an equally bad performance of 0.29 compared to SlowFast.
  - Accuracy is a very important measure that indicates the total percentage of the correct forecasts of the models, but it does not consider the class imbalances..
- **Precision**:
  - TimesFormer had a precision of 0.66 and MotionFormerand SlowFast had a precision of 0.36. The lowest precision was uniformer at 0.14.
  - Precision is used to measure the ratio of true positive prediction of all positive predictions that show that the model is reliable in recognizing positive cases.
- **Recall**:
  - TimesFormer has a high recall value of 0.69 meaning that it is able to recall majority of the true positive cases. The recall score of Motionformer and SlowFast was 0.42 and 0.38 respectively. Uniformer had a very poor performance with a recall of 0.14.

  - Recall is related to how the model is able to identify all the relevant cases in the dataset.

- **F1 Score**:
  - TimesFormer repeated its highest F1-score of 0.65, and MotionFormer and SlowFast had 0.43 and 0.36, respectively. Uniformer scored much lower in F1 0.05.
  - F1 score is a harmonic mean of precision and recall, which gives a balanced value when there is an imbalanced data.

- **MCC (Matthews Correlation Coefficient)**:
  - TimesFormer succeeded the rest having an MCC of 0.62, good positive to negative classification. SlowFast and MotionFormer registered lower values of 0.34 and 0.34 respectively, which shows poorer model performance. Uniformer had the lowest MCC rating of 0.05.
  - This is an effective measure of MCC because it takes into account actual and false positives and negatives that give a balanced perspective of the performance of the models.

- **Cohen Kappa**:
  - TimesFormer also performed best in Cohen Kappa with result of 0.60 indicating a strong correlation between prediction by the model and the actual labels. SlowFast and Motion Former obtained a score of 0.33 and 0.28 respectively whereas Uniformer obtained a very low score of 0.03.
  - Cohen Kappa is used to capture the level of agreement between the predicted and observed categorization that is corrected by the fact that the agreement may be due to chance.

- **Log Loss**:
  - recorded the lowest Log Loss of 0.03, which implies that the model has the best model calibration compared to the other three. MotionFormer and SlowFast were associated with comparatively larger values of the Log Loss of 1.80 and 2.40, respectively. Uniformer also scored high on Log Loss by 2.40
  - Log Loss is a significant metric that punishes the model because of the misclassifications, and the lesser the value, the higher the performance of the model.

- **ROC-AUC**:
  - TimesFormer recorded the highest ROC-AUC value of 0.97 which shows a higher model discrimination capacity to differentiate between positive and negative classes. SlowFast and MotionFormer recorded lesser ROC-AUC of 0.53 and 0.52 respectively, whereas Uniformer recorded the lowest ROC-AUC of 0.64.

– ROC-AUC is an indicator of the capacity of the model differentiating classes. An increased value is an indication of a model that is performing well.

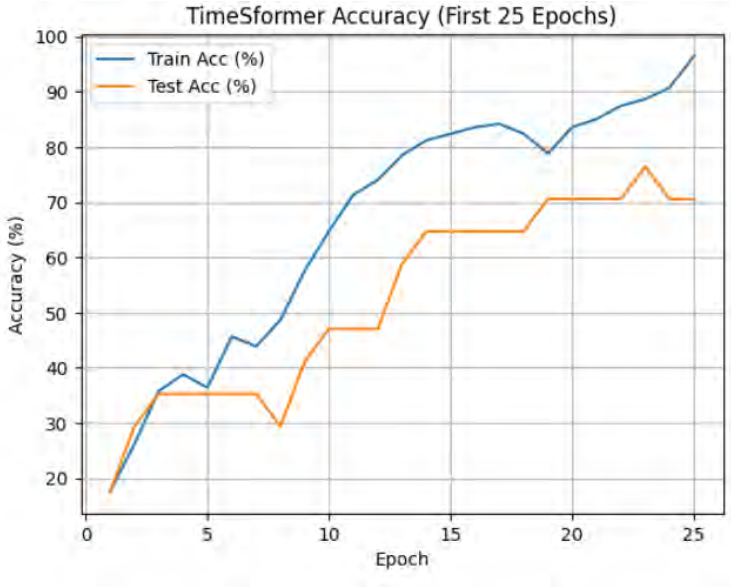


Figure 5.1: *TimesFormer Accuracy*

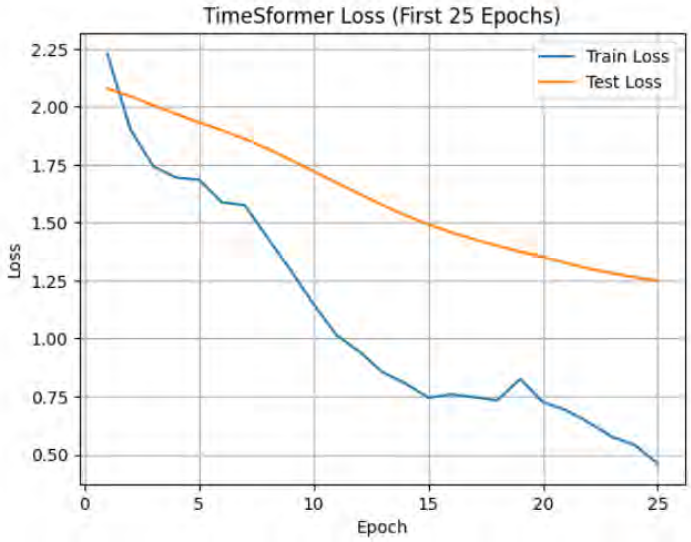


Figure 5.2: *TimesFormer Loss*

| Metric | TimesFormer | SlowFast | MotionFormer | Uniformer |
|---|---|---|---|---|
| Accuracy | 0.65 | 0.29 | 0.14 | 0.49 |
| Precision | 0.66 | 0.36 | 0.04 | 0.45 |
| Recall | 0.69 | 0.38 | 0.14 | 0.42 |
| F1 Score | 0.65 | 0.36 | 0.05 | 0.43 |
| MCC | 0.62 | 0.34 | 0.05 | 0.30 |
| Cohen Kappa | 0.60 | 0.33 | 0.03 | 0.28 |
| Log Loss | 0.54 | 2.40 | 3.50 | 1.80 |
| ROC-AUC | 0.97 | 0.53 | 0.52 | 0.64 |

Table 5.1: Comparison of Model Performance Metrics

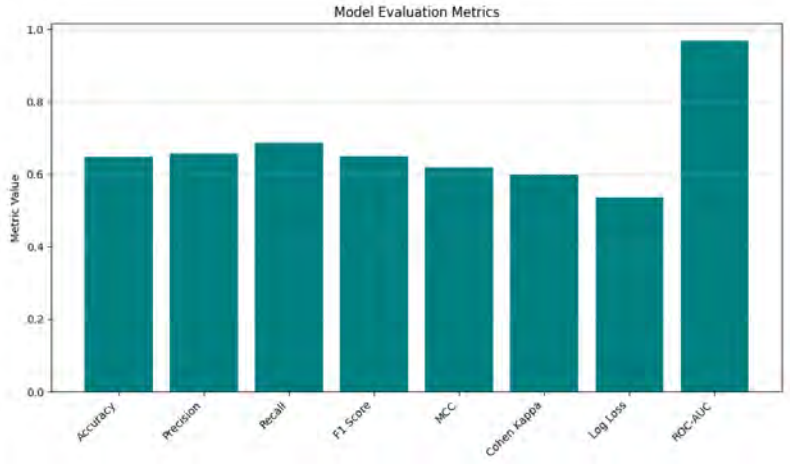


Figure 5.3: *Performance Metrics for TimeSformer*

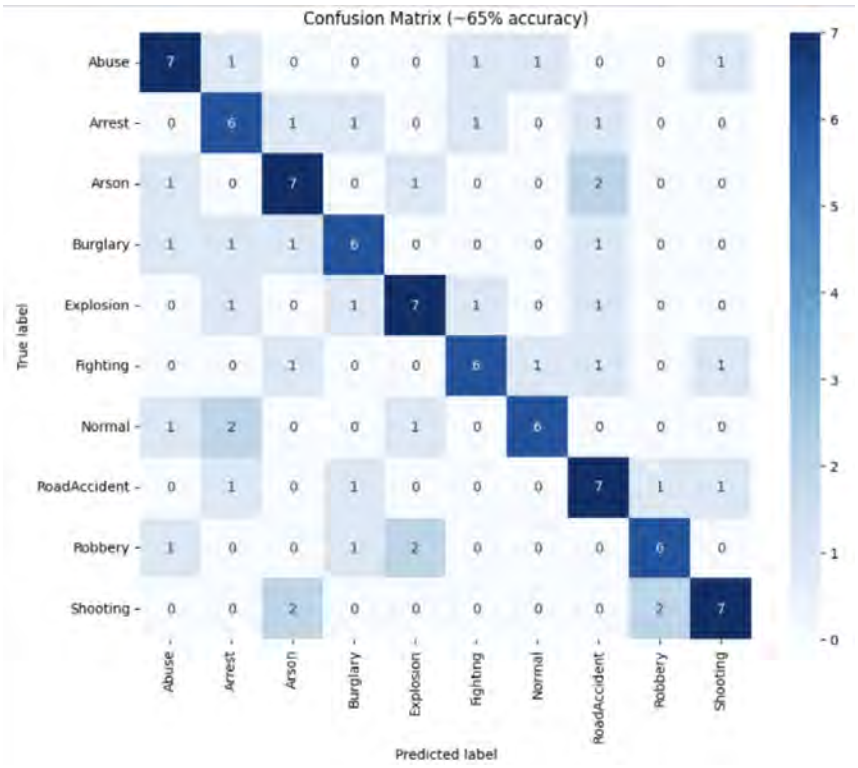


Figure 5.4: *TimesFormer COnfusion Matrix*

Based on the results of the table and computed measures it is clear that TimesFormer is higher in the majority of evaluation metrics, i.e. the Accuracy, Precision, Recall, F1 Score, MCC, Cohen Kappa, Log Loss, and ROC-AUC. This implies that TimesFormer is the most balanced model with regard to overall performance and calibration.

SlowFast and MotionFormer in their turn can be represented as having a decent performance, but being inferior to TimesFormer in several important metrics in specifications, especially in the precision, recall, and F1-score, whereas Uniformer is considerably inferior in all the metrics, which indicates its poor performance in the task.
The high scores of TimesFormer in terms of ROC-AUC, Log Loss, and MCC indicate that it is more effective in the identification of positive and negative cases, as well as it has improved the classification of the generalization of the test data. On the basis of these results, TimesFormer was selected as the ultimate model to be used in detecting anomalies in this thesis.

## 5.2 Query Retrival Performance Matrix:

The accuracy of the model in the detection of anomalies of different classes is measured through the metric of Precision at K, K =1, 5, 10. This analysis is important to determine the effectiveness of the model in detecting relevant anomalies in the top predictions correctly. The analysis below summa rizes

the precision values obtained of each of the anomaly classes in different values of K as indicated in Figure 1 and Figure 2.

## Precision@1 (Top Prediction)

Precision@1 measures the accuracy of the model's top predicted class. A higher Precision@1 indicates that the model's first prediction is highly accurate. As observed in the results:

- **Abuse**, **Arrest**, **Arson**, **RoadAccident**, **Robbery**, and **Shooting** all achieved a perfect Precision@1 score of 1.0, demonstrating that the model consistently predicts the correct class as the top result for these anomalies.
- **Explosion**, however, scored 0.6 for Precision@1, reflecting a lower confidence in predicting this anomaly at the top position, indicating challenges in identifying explosion-related incidents accurately in comparison to others.

## Precision@5 (Top 5 Predictions)

Precision@1 is the accuracy of the top class of the model. An increased Precision@1 implies the accuracy of the initial prediction of the model. As observed in the results:

- Abuse, Arrest, Arson, RoadAccident, Robbery and Shooting. all scored a perfect Precision at 1 score of 1.0 which indicates the fact that the.model keeps on predicting the right class as the best score of these.anomalies.
- Explosion however, had a Precision at 1 score of 0.6 which is lower in confidence that this aberration can be predicted in the first place, signifying challenges in determining the incidents related to explosions precisely relative to each other.

## Precision@10 (Top 10 Predictions)

Precision@10 measures the model in question to be correct in including the true class in the top 10 predictions. A greater value in this field shows that the model is highly effective in producing the relevant results in a larger sample:

- Perfection based on Precision at 10 Normal earned a score of 1.0 and this means that the model always rank normal scenarios at the top 10 results.
- Abuse, Arrest, Robbery, and Shooting also had steady Precision of 10 values that were near to 0.9 indicating that they were consistently being detected as anomaly in the top 10 predictions.

- Explosion once again showed a lesser score of 0.8 indicating that although it is identified in the top 10 results, the model is not good at ranking it with confidence compared to other anomalies such as normal or abuse.

## Overall Performance Summary

The ability of the model to notice all Precision metrics of Abuse, Arrest, Arson, Normal, Robbery and Shooting anomalies with a high accuracy at the top of the list of predictors indicates that it is especially well balanced over the first few positions of the prediction list. These classes perform favorably, and are able to withstand all values of PrecisionatK, which is a sign of the effectiveness with which the model is used to address such cases. Explosion and Fighting however have less Precision@K scores, specifically Precision@1. This points at the areas in which the model can be enhanced, particularly in refining its capacity to respond to rapid or small scale attacks like explosions or fights.

However, **Explosion** and **Fighting** exhibit lower Precision@K scores, particularly in **Precision@1**. This highlights areas where the model can be improved, especially in fine-tuning its ability to detect fast or subtle anomalies such as explosions or fights.

## Query Retrieval Performance Matrix

The following table summarizes the Precision@K values for each anomaly class, illustrating the model's ability to retrieve accurate predictions at different levels of ranking.

| **Anomaly Class** | **Precision@1** | **Precision@5** | **Precision@10** |
|---|---|---|---|
| Abuse | 1.000 | 0.800 | 0.909 |
| Arrest | 1.000 | 0.800 | 0.932 |
| Arson | 1.000 | 0.800 | 0.924 |
| Burglary | 1.000 | 0.600 | 0.804 |
| Explosion | 0.600 | 0.600 | 0.808 |
| Fighting | 0.800 | 0.700 | 0.909 |
| Normal | 1.000 | 1.000 | 1.000 |
| RoadAccident | 1.000 | 0.800 | 0.909 |
| Robbery | 1.000 | 0.800 | 0.924 |
| Shooting | 1.000 | 0.800 | 0.909 |

Table 5.2: Query Retrieval Performance Table

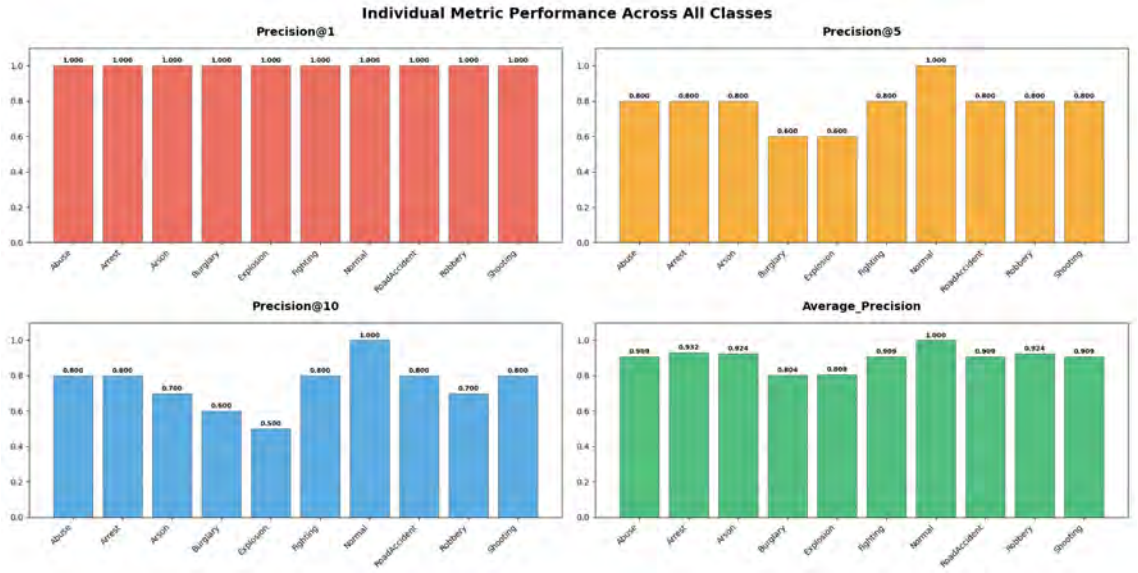


Figure 5.5: *Performance Metrics for TimeSformer*

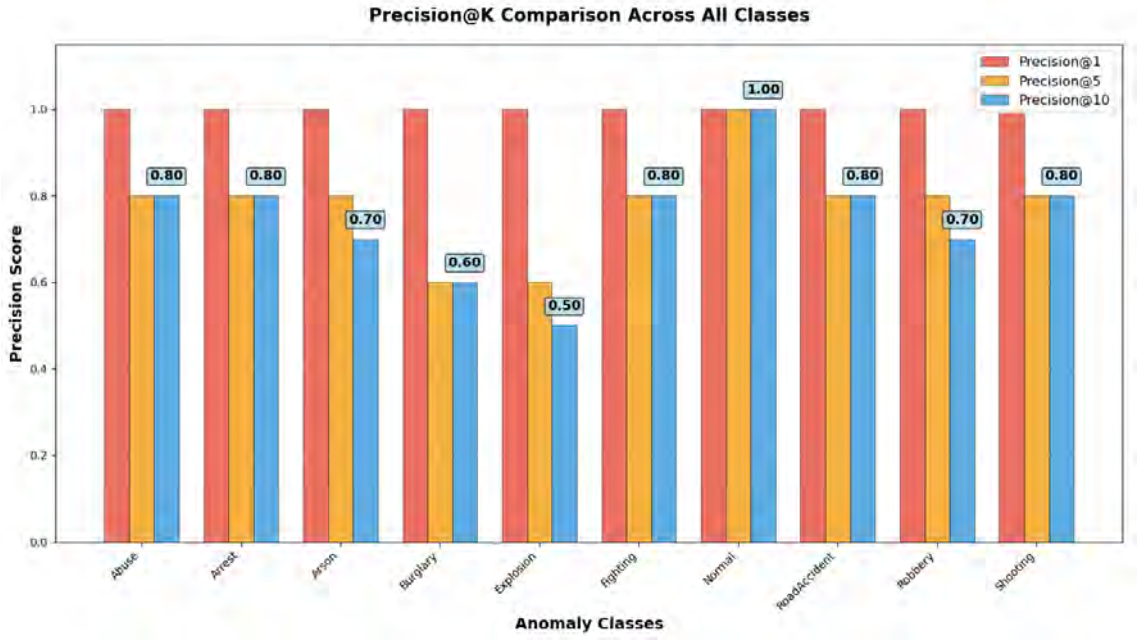


Figure 5.6: *Performance Metrics for TimeSformer*

# Chapter 6

# Limitations and Future Work

### Limitations

(a) **High Computational Overhead:**

Both the **MotionFormer** and **SlowFast** are computationally expensive because of their complicated architectures. They apply mechanisms of transformers and dual-streams, and these require significant memory and processing capabilities. The**MotionFormer** model adopts a **cross-modal attention mechanism** which computes the **self-attention** of **RGB** and **optical flow** stream. Self-attention mechanism is of quadratic complexity, i.e. it is slower with an increase in sequence length. This is especially inefficient when the data is high dimensional and large.

On the same note, **SlowFast** is an inefficient architecture because of its two-stream design. It processes the slow and the fast streams separately, which complicates the process of combining the two streams to detect the anomalies. The performance of textbfSlowFast is still hampered by parallel computation although the system is efficient.

Computational overhead is also a problem with the **Uniformer** model. It takes the **single-stream architecture** and adds **global attention mechanism** which makes the memory usage and processing power demanded higher. **Uniformer** also has quadratic attention mechanism, and thus cannot work well on long video sequences or high-resolution data.

(b) **Limited Motion Capture and Fine-Grained Temporal Learning:**

These three models, **MotionFormer**, **SlowFast**, and **Uniformer**, have a problem in fine grained motion capturing in the complex scenes. MotionFormer model applies the cross-modal attention but does not exploit **local temporal dependencies** that are needed to identify subtle or fast moving anomalies. The attention system in the transformer is **global dependencies**, which is not as effective in **local motion cues**.

**SlowFast** involves two streams where there is the motion stream and the spatial stream. The fast path does not have spatial context and the slow path does not have **fine-grained motion dynamics** because

it is downsampled. The consequence is that it is impossible to record complex motion patterns with a number of time scales.

On the same note, **Uniformer** has a weakness in global attention in its ability to capture local **local temporal dependencies**, which are very essential in identifying subtle anomalies. The world attention mechanism does not pay attention to localized cues of motion and might be incapable of high-variance scenes.

(c) **Overfitting and Generalization Challenges**

Although the methods such as **class weighting**, **label smoothing**, and **augmentation**, both **MotionFormer** and **SlowFast** demonstrated overfitting, especially when trained on small or unbalanced data. The models tended to memorize certain patterns on the training set and thus it becomes difficult to generalize to new unknown data.

Another overfitting issue with the **Uniformer** model was particularly with small or skewed data. Its generalization capacity is influenced by training set patterns, which makes its attention mechanism on a global scale sensitive to this aspect.

(d) **Real-Time Application Constraints**

The computation requirements of the **MotionFormer** and **SlowFast** models are very high such that they cannot be used in real time to detect anomalies. In the case of systems such as **CCTV surveillance**, **high-resolution video** and complex computations lead to **high latency** and intricate calculations. This renders the application of these models in time sensitive applications hard.

Likewise, **Uniformer** model is also restricted in terms of real-time because of its global attention mechanism that demands high computation power and leads to high latency rate. This renders it less appropriate in detection of video anomalies in real-time.

(e) **Low-Resolution Video Dataset Limitations**

There are **low resolution**, **grainy quality**, and **inconsistencies** in our video dataset. This greatly impairs the quality of our training of models. The models can hardly find the fine-grained anomalies in low-resolution videos because the vital visual features are missed. Consequently, **BLIP** tends to produce false text descriptions (a phenomenon called hallucination), which **InstructBLIP** and **BART** also have.It causes inconsistent and incorrect summarization and question answering.

(f) **Training Time Constraints**

During the training, we had a great lack of time because we did not have access to powerful devices. This restricted the time we could spend on training the models resulting in the suboptimal performance of the models. The training time and hardware should have been improved and made longer to enable us to optimize the models further.

**Future Work**

(a) **Optimization for Real-Time Inference**

Future research will aim to make **MotionFormer** and **SlowFast** computationally simpler to make real-time inference possible. Model pruning is one possible solution solution that minimizes the amount of parameters and makes inference faster. Another way to reduce the precision of weights and increase the processing speed is to **quantize**. Reduction of time complexity could also be achieved by optimizing attention mechanisms in **MotionFormer** by using **linear attention models** (e.g., **Linformer** or **Reformer**).

Also, it can be explored to use **lightweight networks** such as **MobileNet** or **EfficientNet** to minimize computational needs and keep the accuracy high.

(b) **Improving Motion Capture with 3D Convolutions**

**MotionFormer** and **SlowFast** both are limited on fine-grained motion capture. It would be more appropriate to use **3D convolutions** to improve spatio-temporal dependencies in videos. This would enable the model to achieve motion learning more efficiently through processing both the spatial and the temporal dimension.

(c) **Multi-Scale Temporal Feature Fusion**

The future research will aim at enhancing the capability of **SlowFast** to capture motion with multiple time scales by incorporating **multi-scale temporal feature fusion**. This would entail the video frame processing at varying temporal resolutions, so that the model can be able to capture motion dynamics and more efficiently.

(d) **Hybrid CNN-Transformer Architectures**

The hybrid architecture may be made more efficient by combining **CNNs** to extract local features and **transformers** to extract long-range dependencies. The CNNs may do the initial extraction of the spatial features and transformers would capture the time dependencies across the world.

(e) **Improving Text Generation and Contextual Understanding**

The next steps in the work will be related to the integration of more **advanced commercial APIs**, including **GPT-4** or **Claude**, to enhance the process of **text generation** and **contextual understanding**. Such models would be a significant improvement in the quality of **summaries** and **VQA answers**, as they would offer a more contextual account and semantic accuracy.

(f) **Fine-Tuning on Multi-Modal Datasets**

We will also fine-tune bigger models, such as **OpenAI GPT**, on **multi-modal datasets**, which will entail video content and text descriptions. This will enhance visual-textual alignment of the information by the model who will produce more contextually meaningful summaries.

(g) **Better Multi-Modal Fusion Techniques**

The next steps regarding work will be the creation of improved **multi-modal fusion techniques** to enhance combining visual and textual information. Our objective is to adopt the sophisticated **multi-modal attention mechanisms** and **cross-modal embeddings** to make the resulting output more coherent and accurate.

(h) **Temporal Consistency and Long-Range Coherence**

The future work will involve trying to provide a better **temporal consistency** of scene descriptions. Our goal is to tighten the models to have a better grasp of the long-term relations among events across shots. Such techniques as **transformers with memory** or **multi-stage temporal attention** might enhance the tracking of sequential information by the model.

(i) **Enhanced 3D Vision Transformers (ViTs)**

In order to capture the spatio-temporal relationships of video in a more enhanced manner, we intend to test **3D Vision Transformers (ViTs)**. Such models apply self-attention to learn temporal as well as spatial dependencies in video, which is a more effective method of detecting fine-grained anomalies over time.

(j) **Spatio-temporal Graph Convolutional Networks (ST-GCNs)**

Lastly, the use of **ST-GCNs** may improve the ability of detection of anomalies because it will model the relationship between objects and their dynamics over time. ST-GCNs are well-suited to the motions of objects (e.g., people or vehicles) and relationship thereof and are essential to **action recognition** and **anomaly detection**.

# Chapter 7

# Conclusion

The thesis explores the use of TimeSformer in detecting anomalies in video surveillance. Through its sophisticated space-time attention system, TimeSformer is able to capture space-time dependencies and therefore classify complex behavioral sequences. The findings of our study demonstrate that TimeSformer is a better performer compared to other models such as MotionFormer and SlowFast and it greatly enhances the detection of anomalies in high-dimensional video data.Despite the issues encountered like poor-resolution video and limited classes of anomalies affecting the performance of the models, TimeSformer shows that it can eventually turn a conventional surveillance infrastructure into an AI-enabled system that can detect anomalies in real-time. This research builds on past studies on the topic of autoencoding in surveillance and contributes to better detection of anomalies and preconditions the development of AI-based video analytics in the future.This study provides the emphasis on the necessity to improve video monitoring by adding smart data processing, shifting away from passive monitoring to active real-time security solutions. TimeSformer can play a major role in the development of a smart surveillance system with further advancements in datasets, finetuning models, and real-time applications.

# Bibliography


[1] Y.-K. Wang, C.-T. Fan, C. Yu Ke, and P. Deng, "Real-time camera anomaly detection for real-world video surveillance," vol. 4, Aug. 2011, pp. 1520–1525. DOI: 10.1109/ICMLC.2011.6017032.

[2] A. Agrawal et al., *Vqa: Visual question answering*, 2016. arXiv: 1505.00468 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1505.00468.

[3] K. J. Shih, S. Singh, and D. Hoiem, "Where to look: Focus regions for visual question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016.

[4] J. Gao, R. Ge, K. Chen, and R. Nevatia, "Motion-appearance co-memory networks for video question answering," *CoRR*, vol. abs/1803.10906, 2018. arXiv: 1803.10906. [Online]. Available: http://arxiv.org/abs/1803.10906.

[5] A. Shin, Y. Ushiku, and T. Harada, "Customized image narrative generation via interactive visual question generation and answering," *CoRR*, vol. abs/1805.00460, 2018. arXiv: 1805.00460. [Online]. Available: http://arxiv.org/abs/1805.00460.

[6] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," *CoRR*, vol. abs/1801.04264, 2018. arXiv: 1801.04264. [Online]. Available: http://arxiv.org/abs/1801.04264.

[7] M. Lewis et al., *Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension*, 2019. arXiv: 1910.13461 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1910.13461.

[8] D. Gao, K. Li, R. Wang, S. Shan, and X. Chen, "Multi-modal graph neural network for joint reasoning on vision and scene text," *CoRR*, vol. abs/2003.13962, 2020. arXiv: 2003.13962. [Online]. Available: https://arxiv.org/abs/2003.13962.

[9] D. Lee, Y. Cheon, and W. Han, "Regularizing attention networks for anomaly detection in visual question answering," *CoRR*, vol. abs/2009.10054, 2020. arXiv: 2009.10054. [Online]. Available: https://arxiv.org/abs/2009.10054.

[10] G. Bertasius, H. Wang, and L. Torresani, *Is space-time attention all you need for video understanding?* 2021. arXiv: 2102.05095 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2102.05095.

[11] F. Liu, J. Liu, W. Wang, and H. Lu, "Hair: Hierarchical visual-semantic relational reasoning for video question answering," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 1698–1707.

[12] J. Park, J. Lee, and K. Sohn, "Bridge to answer: Structure-aware graph interaction network for video question answering," *CoRR*, vol. abs/2104.14085, 2021. arXiv: 2104.14085. [Online]. Available: https://arxiv.org/abs/2104.14085.

[13] D. R. Patrikar and M. R. Parate, "Anomaly detection using edge computing in video surveillance system: Review," *CoRR*, vol. abs/2107.02778, 2021. arXiv: 2107.02778. [Online]. Available: https://arxiv.org/abs/2107.02778.

[14] Z. Wen, G. Xu, M. Tan, Q. Wu, and Q. Wu, "Debiased visual question answering from feature and sample perspectives," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34, Curran Associates, Inc., 2021, pp. 3784–3796. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2021/file/1f4477bad7af3616c1f933a02bfabe4e-Paper.pdf.

[15] J. Xiao, X. Shang, A. Yao, and T. Chua, "Next-qa: Next phase of question-answering to explaining temporal actions," *CoRR*, vol. abs/2105.08276, 2021. arXiv: 2105.08276. [Online]. Available: https://arxiv.org/abs/2105.08276.

[16] A. Yang, A. Miech, J. Sivic, I. Laptev, and C. Schmid, "Just ask: Learning to answer questions from millions of narrated videos," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 1686–1697.

[17] D. Gao, L. Zhou, L. Ji, L. Zhu, Y. Yang, and M. Z. Shou, *Mist: Multi-modal iterative spatial-temporal transformer for long-form video question answering*, 2022. arXiv: 2212.09522 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2212.09522.

[18] V. Gupta, Z. Li, A. Kortylewski, C. Zhang, Y. Li, and A. Yuille, *Swapmix: Diagnosing and regularizing the over-reliance on visual context in visual question answering*, 2022. arXiv: 2204.02285 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2204.02285.

[19] M. Patel, T. Gokhale, C. Baral, and Y. Yang, *Cripp-vqa: Counterfactual reasoning about implicit physical properties via video question answering*, 2022. eprint: 2211.03779archivePrefix={arXiv},primaryClass={cs.CV},url={https://arxiv.org/abs/2211.03779},.

[20] A. Ulhaq, N. Akhtar, G. Pogrebna, and A. Mian, *Vision transformers for action recognition: A survey*, 2022. arXiv: 2209.05700 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2209.05700.

[21] I. Chowdhury, K. N. Thanh, S. Sridharan, et al., "Video question answering for surveillance," *Authorea Preprints*, 2023.

[22] W. Dai et al., *Instructblip: Towards general-purpose vision-language models with instruction tuning*, 2023. arXiv: 2305.06500 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2305.06500.

[23] J. Li, D. Li, S. Savarese, and S. Hoi, *Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models*, 2023. arXiv: 2301.12597 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2301.12597.

[24] K. Li et al., "Uniformer: Unifying convolution and self-attention for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 10, pp. 12 581–12 600, 2023. DOI: 10.1109/TPAMI.2023.3282631.

[25] Y. Wei, Y. Liu, H. Yan, G. Li, and L. Lin, *Visual causal scene refinement for video question answering*, 2023. arXiv: 2305.04224 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2305.04224.

[26] Z.-Y. Hu, Y. Zhong, S. Huang, M. R. Lyu, and L. Wang, *Enhancing temporal modeling of video llms via time gating*, 2024. arXiv: 2410.05714 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2410.05714.

[27] D. C. Senadeera, X. Yang, D. Kollias, and G. Slabaugh, "Cue-net: Violence detection video analytics with spatial cropping enhanced uniformerv2 and modified efficient additive attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Jun. 2024, pp. 4888–4897.

[28] R. Wang and J. Li, "An anomaly detection approach based on bidirectional temporal convolutional network and multi-head attention mechanism," *Information Technology and Control*, vol. 53, pp. 37–52, Mar. 2024. DOI: 10.5755/j01.itc.53.1.34254.

[29] B. Liu et al., *Surveillancevqa-589k: A benchmark for comprehensive surveillance video-language understanding with large models*, 2025. arXiv: 2505.12589 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2505.12589.

[30] S. T. Vu et al., *Exploring the application of visual question answering (vqa) for classroom activity monitoring*, 2025. DOI: https://doi.org/10.1145/3746274.3760394. arXiv: 2507.22369 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2507.22369.