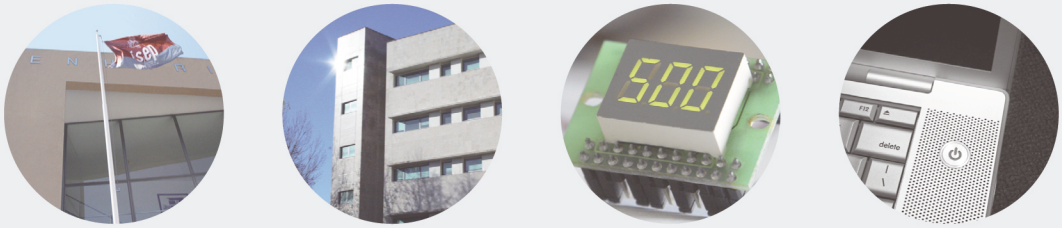
INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO
MESTRADO EM ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES



Analysing the Impact of Emerging Backbones on Generalization of Video Anomaly Detection Models

PAULO MIGUEL BORGES SILVA
outubro de 2024

POLITÉCNICO
DO PORTO

Politécnico do Porto
Instituto Superior de Engenharia do Porto

Analysing the Impact of Emerging Backbones on Generalization of Video Anomaly Detection Models

Paulo Miguel Borges Silva

Master in Electrical and Computer Engineering
Specialization Area of Automation and Systems

Departamento de Engenharia Eletrotécnica
Instituto Superior de Engenharia do Porto

October, 2024

This dissertation partially satisfies the requirements of the Thesis/Dissertation course of the program Master in Electrical and Computer Engineering, Specialization Area of Automation and Systems.

Candidate: Paulo Miguel Borges Silva, No. 1190937, 1190937@isep.ipp.pt

Scientific Guidance: Dr. Pedro Miguel Carvalho, pedro.m.carvalho@inesctec.pt

Company: INESC-TEC

Advisor: Dr. Pedro Miguel Carvalho, pedro.m.carvalho@inesctec.pt

Departamento de Engenharia Eletrotécnica
Instituto Superior de Engenharia do Porto
Rua Dr. António Bernardino de Almeida, 431, 4200–072 Porto

October, 2024

Abstract

Video anomaly detection plays a crucial role in intelligent surveillance systems, where it is essential to identify events that deviate from normal behaviour. These systems offer several advantages such as real-time monitoring that allows immediate response to security threats, scalability to process large volumes of data across different environments and it ensures anomalies are detected without being influenced by human error or human corruption, highly affected in areas like public spaces and prisons.

A significant challenge consists in achieving strong Out-of-Distribution generalization, which ensures models perform effectively on unseen data. This dissertation investigates the impact of emerging backbone architectures and advanced learning techniques on the performance of the models, with a focus on improving their ability to generalize across varied and complex real-world scenarios.

The study offers a comprehensive comparison of backbone architectures, ranging from traditional to cutting-edge, for the task of anomaly detection. Additionally, it examines the potential of Self-Supervised Learning methods to overcome the limitations of conventional supervised approaches, particularly in improving generalization across diverse datasets. On the other hand, recent literature on Semi-Supervised models indicates that novel backbones do not show significant improvements. However, leveraging One-Class Classification methods may offer better generalization.

The findings reveal that multi-modal self-supervised backbones, such as Contrastive Language-Image Pretraining, demonstrate strong performance in anomaly detection even performing novelty detection, however single-modal techniques like Self-Distillation with No Labels are highly sensitive to scenario conditions. Hybrid architectures like NextViT exhibit limited advancements over existing solutions. Additionally, One-Class Classification methods have proven to be effective in controlled environments with minimal variations, offering a simpler and more robust alternative to complex approaches and backbones.

Keywords: Video Anomaly Detection, Deep Learning, Computer Vision, Benchmark.

Resumo

A Deteção de Anomalias em Vídeo desempenha um papel crucial em sistemas inteligentes de vigilância e monitorização, sendo essencial para identificar eventos que se desviam do comportamento normal. Estes sistemas oferecem vantagens como automação e escalabilidade, permitindo operar simultaneamente em múltiplas zonas, eliminando a necessidade de intervenção humana e a fadiga associada. Além disso, reduzem os custos relacionados à videovigilância, ao diminuir o tamanho das equipas necessárias, e eliminam o risco de corrupção, um problema inerente à natureza humana.

Logo à partida trás vantagens como automação e escalabilidade visto que estes modelos podem operar em múltiplas zonas simultaneamente eliminando a necessidade da intervenção humana e as consequências que poderão daí advir como fadiga; reduz os custos associados à videovigilância já que não é necessário depender de grandes equipas de videovigilância e reduz o risco de corrupção que é um problema que se prende com a natureza humana.

Um desafio significativo consiste em alcançar uma forte generalização *Out-Of-Distribution*, assegurando que os modelos funcionem de forma eficaz para dados nunca vistos. Esta dissertação investiga o impacto das arquiteturas de *backbone* emergentes e das técnicas de aprendizagem avançadas no desempenho dos modelos, com foco na melhoria da sua capacidade de generalização em cenários variados e complexos do mundo real.

Este estudo avalia o desempenho das arquiteturas tradicionais de *Convolutional Neural Networks*, *Transformers* e arquiteturas híbridas em tarefas de deteção de anomalia. Explora também o papel dos métodos de Aprendizagem Auto-Supervisionada, que estão a tornar-se cada vez mais relevantes em modelos de Supervisão Fraca, em comparação com as abordagens supervisionadas tradicionais. Por outro lado, a literatura recente sobre modelos Semi-Supervisionados indica que os *backbones* novos não mostram melhorias significativas. No entanto, é possível tirar proveito dos métodos de One-Class Classification para uma melhor generalização.

Os resultados revelam que os *backbone* auto-supervisionados multimodais, como o Contrastive Language-Image Pretraining, demonstram um forte desempenho na deteção de anomalias, incluindo a deteção de novidades, no entanto técnicas unimodais como Self-Distillation with No Labels são altamente sensíveis às condições do cenário. Modelos de arquitetura híbrida, como o NextViT, apresentam avanços

limitados em relação às soluções existentes. Além disso, os métodos *One-Class Classification* provaram ser eficazes em ambientes controlados com variações mínimas, oferecendo uma alternativa mais simples e robusta aos *backbone* complexos.

Palavras-Chave: Deteção de Anomalias de Vídeo, *Deep Learning*, Visão Computacional, *Benchmark*.

Contents

List of Figures

List of Tables

List of Acronyms

AI	Artificial Inteligence
AP	Average Precision
AUC	Area Under Curve
C3D	Convolutional 3D
CDE	Cross-Dataset Evaluation
CLIP	Contrastive Language-Image Pretraining
CNN	Convolutional Neural Network
DINO	Self-Distillation with No Labels
E-MHSA	Efficient Multi-Head Self Attention
FAR	False Alarm Rate
FPR	False Positive Rate
GELU	Gaussian Error Linear Unit
I3D	Inflated 3D
IDE	Intra-Dataset Evaluation
LDA	Linear Discriminant Analysis
MHCA	Multi-Head Convolutional Attention
MHSA	Multi-Head Self Attention
MLP	Multilayer Perceptron
NCB	Next Convolution Block
NLP	Natural Language Processing
NTB	Next Transformer Block
OCC	One-Class Classification

OOD	Out-Of-Distribution
P-R	Precision-Recall
PCA	Principal Component Analysis
PDC	Pyramid Dilated Convolutions
PSNR	Peak Signal-to-Noise Ratio
ResNet	Residual Neural Network
RNN	Recurrent Neural Networks
ROC	Receiver Operation Characteristic
SOTA	State-of-the-art
SS	Semi-Supervised
SSL	Self-Supervised-Learning
TPR	True Positive Rate
VAD	Video Anomaly Detection
ViT	Vision Transformer
VOD	Video Object Detection
WS	Weakly-Supervised

Chapter 1

Introduction

This chapter identifies the challenges of anomaly detection, its real-world impact, approaches to the problem, and limitations in model generalization. It also presents the objectives, timeline, and document structure.

1.1 Context and Motivation

Video Anomaly Detection (VAD) is a critical research area within artificial vision, especially in the realm of intelligent surveillance systems. Currently, most video surveillance systems require manual monitoring by security personnel, leading to challenges like operator fatigue during extended shifts. The primary goal of VAD models is to automatically identify events in video footage that deviate from normal behavior such as thefts, robberies, explosions, or traffic accidents. This capability is essential for maintaining safety and security across various environments.

VAD covers a wide range of real-world situations such as public safety, store security, accident detection and healthcare environments. Some of the advantages may include [1]:

- Automation and Scalability: VAD models can monitor multiple video streams simultaneously eliminating the need for human operators to constantly watch footage;
- Early Detection and Prevention: VAD systems can detect unusual behavior in real-time, enabling security teams or personnel to respond faster;

- Cost Efficiency and Less Corruption: Since VAD models do not rely on human judgment, it reduces the risk of intentional oversight or bias, ensuring that all anomalies are detected and flagged impartially. Beyond that, it reduces labor costs associated with employing large teams of human operators to monitor video feeds.

Detecting anomalous events is inherently complex due to their unpredictable nature. The growing volume of video data available makes manual processing of this data almost impossible and it is necessary to have this data properly labeled in order to train anomaly detection models. However, this labelling effort did not led to any improvement in generalization of this models.

Emerging backbone architectures and advanced learning techniques have significantly advanced image processing in fields like Video Object Detection (VOD) and Image Classification, surpassing outdated backbones still highly leveraged by VAD models. As a result, integrating these modern feature extraction methods into VAD presents a promising opportunity to enhance performance and generalization.

1.2 Types of Approach in VAD

Over the years, various methods have been developed to tackle the problem of anomaly detection in video data. These approaches range from Fully-Supervised methods, which require detailed frame-level labeling, to Weakly-Supervised (WS) methods that operate with abnormal and normal video-level labels, and Semi-Supervised (SS) methods that train models using only normal data. There is also an Unsupervised approach that avoids labeling entirely during training and testing. The Figure 1.1 shows this 4 popular approaches towards anomaly detection [2] [3].

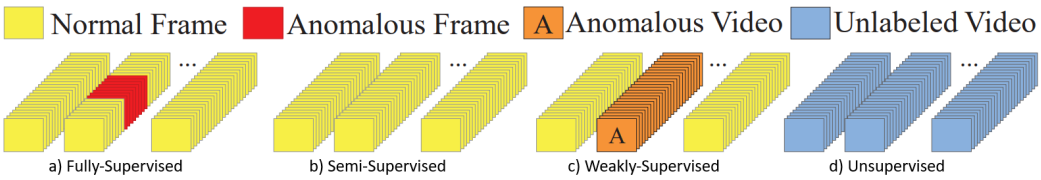


Figure 1.1: Different training modes for each VAD approach [3].

Among these, SS and WS approaches are currently the most prominent. WS methods, in particular, have gained popularity due to their superior performance metrics, such as the area under the Receiver Operation Characteristic (ROC) curve. However, a recent study suggested that SS methods might be more effective in detecting truly novel anomalies because they do not rely on predefined categories of abnormal behavior during training [4].

1.3 Challenges in Model Training and Generalization

Training a VAD model typically involves using a labeled dataset to guide the neural network in distinguishing between normal and abnormal instances. However, due to the vast diversity of scenarios in which anomalies can occur, achieving effective generalization has become increasingly difficult over the years.

A significant challenge in this field is the Out-Of-Distribution (OOD) generalization, which refers to the model's ability to perform well on new data that comes outside its Training Domain. OOD generalization is crucial for VAD models to be reliable in real-world applications, where they will inevitably encounter scenarios not covered in the training data. There are several types of OOD data [5]:

- In-distribution data: this refers to the Testing Set that directly corresponds to the Training dataset;
- Covariate Shift: this refers to changes in image characteristics, such as brightness or contrast, while core features remain unchanged. This shift challenges the model's ability to maintain performance under varying conditions;
- Novelty Detection: this involves encountering images outside the training domain, which is identifying completely new types of anomalies;
- Adversarial Attacks: these are deliberate and malicious changes that trigger false positives and exploit vulnerabilities of the model. Abnormal movements like a person falling or fighting might be manipulated for this purpose.

State-of-the-art (SOTA) models exhibit poor generalization and current backbones are not capable of mitigating Covariate Shift and Adversarial Attacks, Novelty Detection may require a larger increase in models' Training Domain as well as the implementation of new backbone architectures or different learning techniques to be detected. So far, authors have just stayed focused on shrinking In-distribution errors rather than addressing OOD challenges [5].

Figure 1.2 emphasizes the 2 main paradigms of current VAD models: being able to detect data samples outside In-Distribution domain (covered in Model Generalization) while continuing to improve the detection of data samples within the Training Domain (covered in In-distribution errors), with most recent works only focusing on this last one.

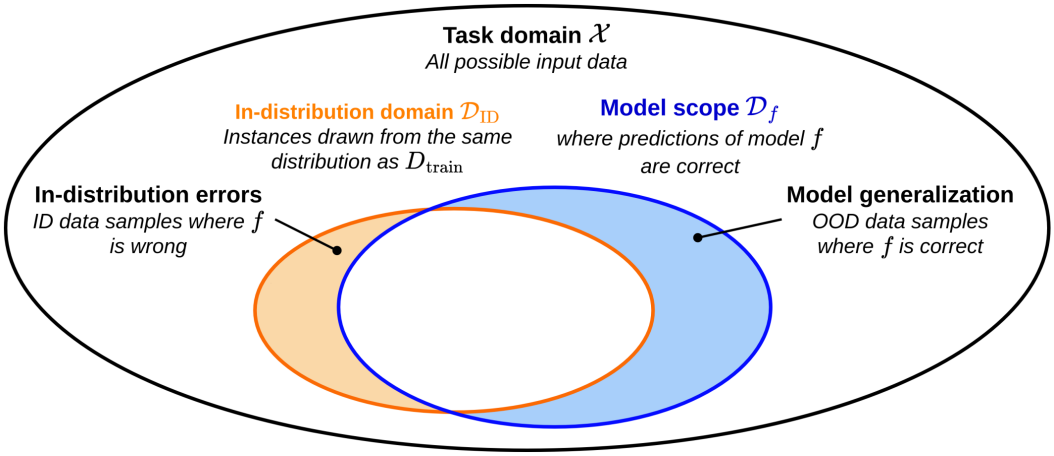


Figure 1.2: Schematic representation of the In-Distribution Domain and the Model Scope of a model f for this task [5].

1.4 Objectives

The primary focus of this work is to analyze the impact of emerging backbone architectures on the generalization capabilities of VAD models. Traditional VAD models rely on backbones that use obsolete learning techniques, limiting the capture of anomalies in scenarios not encountered during training.

In response, this dissertation explores Self-Supervised-Learning (SSL) backbones, specifically Contrastive Language-Image Pretraining (CLIP) and Self-Distillation with No Labels (DINO), along with innovative architectures like NextViT, as alternatives to conventional backbones in WS models. This exploration aims to identify the factors contributing to the varying performance of these backbones.

The second goal is to analyze the role of SS models on generalization, since this type of models do not seem to benefit from novel backbones. Lastly, this work prepares and labels 4 Testing Sets, which represent 4 different scenarios, to evaluate the models under various conditions and analyze their behavior under the new backbones.

1.5 Work Calendarization

The work was developed throughout 33 weeks and divided in 4 phases, as seen in Figure 1.3. Initially, it was made a research on VAD methods and SOTA approaches. Then it took almost a month to install all requirements to run the models as well as CUDA packages. After running the models with open-source code successfully, it is made a deep research on new emerging backbones and recent benchmarks regarding image processing area and anomaly detection. Consequently, after setting up the environment to use novel backbones, the target datasets were prepared and processed

to be fed into the models, which took a long time due to high computational costs and massive data. After cutting all videos and having their features extracted, the models were trained and tested across different backbones within Intra-Dataset Evaluation and Cross-Dataset Evaluation. The results were analysed and noted for this document.

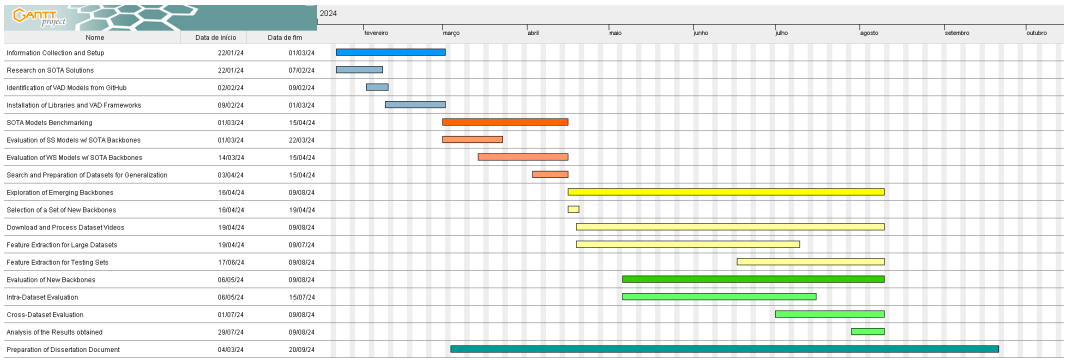


Figure 1.3: Calendarization.

1.6 Document Organization

The document is organized in 7 different chapters: the first chapter introduces the problem of anomaly detection, the motivation for this work and the existent challenges, the second chapter contextualizes the current state of VAD backbones for each approach and describes the new architectures and techniques, the third chapter explains the idea of using new backbones, identifies the baseline models and presents how experiments will be employed, fourth chapter characterizes relations between datasets and identifies the metrics, fifth chapter presents the results obtained, sixth chapter discusses about the results and compares the enhanced and SOTA methods, and chapter seventh summarizes the conclusions obtained from this study.

Chapter 2

Related work

In this chapter it is provided a clear overview on the feature extraction process, composed by 2 important steps: Feature Extraction and Feature Selection. A review of the backbones used in both WS and SS approaches is done, and the new backbones to be tested in this dissertation are described.

2.1 Image Processing in VAD

In real life, all the data that is collected exists in large amount especially in large scale datasets that contain videos of long duration. To understand this data, it is used the concept of features extraction so that Artificial Inteligence (AI) models gain the perception of the real world to perform tasks like anomaly detection [6].

2.1.1 Feature Extraction Process

Feature extraction is a part of the dimensionality reduction process, where raw data is divided and reduced into more manageable groups. Large datasets typically contain a high number of variables, with each image represented by an image embedding, a numeric representation that encodes its semantic content. By selecting and combining relevant variables, feature extraction reduces the data while preserving essential information [6].

A VAD model requires a video processing backbone that processes several key steps to transform raw image data into meaningful features: Image Loading, Pre-Processing, Feature Extraction, Feature Selection and Feature Encoding. The extracted and processed features are used to train or evaluate the AI model [7].

The 2 most important steps in this process are the Feature Extraction and Feature Selection. VAD models leverage Learning-Based methods, based on Neural Networks, to capture high-level and low-level feature types. Linear methods like Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and SelectKBest fail to capture non-linear relationships between features so they are not viable for anomaly detection task.

The following subsection introduces structure concepts of Vision Transformer (ViT), Convolutional Neural Network (CNN) and AutoEncoders. According to literature, Autoencoders can be considered as a Feature Selection structure while ViT and CNN are Feature Extractors.

2.1.2 Visual Transformers

The CNN models are currently dominating the field of computer vision, specifically in the VAD topic. However, recently Visual Transformer models emerged as a real competitive alternative to the CNN since big enterprises like Google, META and OpenAI continue to develop new models that increasingly become more detailed and precise.

The birth of Vision Transformers began in 2017 when a paper entitled "Attention is all you need" pointed out the excellent performance of attention mechanisms with Recurrent Neural Networks (RNN) for facing any task involving text as Natural Language Processing (NLP). At this time, the big challenge was to make this attention mechanisms to efficiently handle large inputs and outputs such as images [8].

In 2021 was published a conference research paper entitled "An Image is Worth 16*16 Words: Transformers for Image Recognition at Scale" which introduced the first ViT model. Figure 2.1 shows a Vision Transformer architecture for Image Classification, which adds a final Multilayer Perceptron (MLP) classification layer.

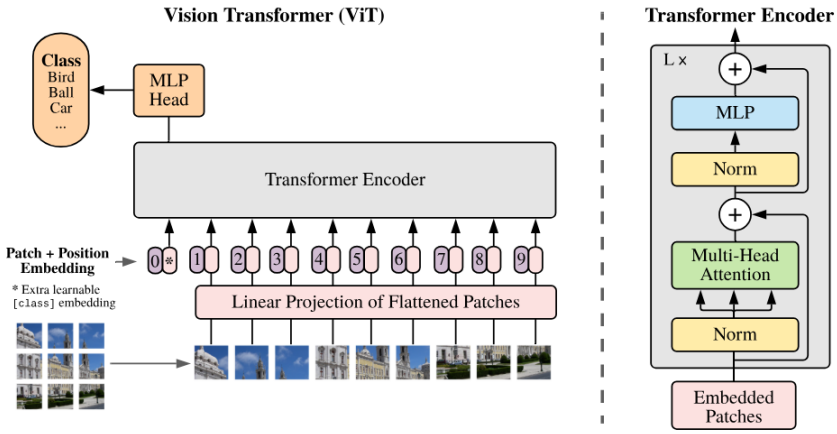


Figure 2.1: Generic Vision Transformer Architecture for Image Classification [9].

The steps of a Vision Transformer architecture can be seen in Figure 2.1 and consist in the following steps [10]:

1. Split an image into patches;
2. Flatten the patches;
3. Produce lower-dimensional linear embeddings from the flattened patches;
4. Add positional embedding;
5. Feed the sequence as an input to a standard transformer encoder;
6. Initially train the model using image labels (fully supervised on a huge dataset or self-supervised in more recent models);
7. Finetune on the downstream dataset for imagem recognition tasks such as image classification, object detection, image segmentation, and action recognition.

The are multiple blocks in the Transformer Encoder (Figure 2.1), and each block consists of 3 major processing elements: Layer Norm, Multi-Head Attention Block and MLP [10]:

- Layer Norm keeps the training process on track and lets the model adapt to different training images;
- Multi-head Attention Network is a network responsible for generating attention maps from the given embedded visual tokens. These attention maps help the network focus on the most critical regions in the image, such as objects;

- MLP is a two-layer classification network with a Gaussian Error Linear Unit (GELU) at the end. The final MLP is also used as an output of the transformer with a final softmax layer to provide classification labels as seen in Figure 2.1.

2.1.3 Convolutional Neural Networks

CNN models are a part of learning-based methods that have shown exceptional performance in image analysis, video processing and computer vision niches. Nowadays, this is the most used approach in VAD solutions. The architecture of a CNN is designed to take advantage of the 2D structure of an input image. This is achieved with the use of feature extraction, a special type of layer, a convolutional layer, and pooling layers. Figure 2.2 shows the basic composition of a CNN architecture [11].

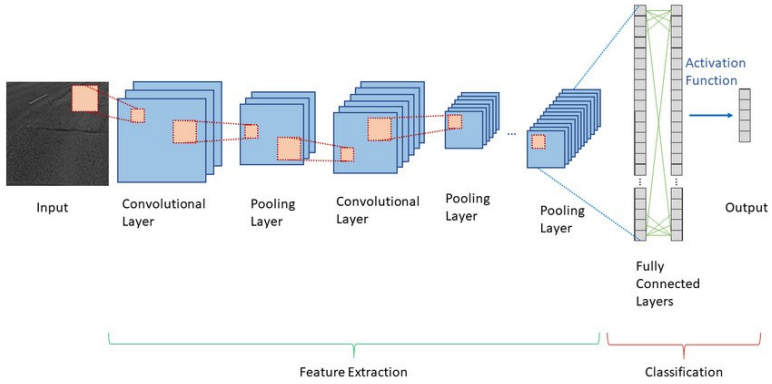


Figure 2.2: Basic Architecture of CNN [11].

The primary function of the convolutional layer is to identify and respond to local patterns or features presented in the previous layer. This is done by processing a three-dimensional input tensor and creating an output tensor through a set of adaptive filters or kernels. Each filter scans over the width and height of the input tensor and the outcome is a two-dimensional activation map that reveals how the filter responds at every spatial location. As the network learns, filters are tuned to activate upon recognizing certain visual characteristics such as an edge or a specific color contrast [11].

Figure 2.3 shows an example of this convolutional process, where a filter (depicted in green) of size 2x2 aligns with a matching area (represented in orange) within a (4x4) input feature map.

Pooling layers are usually inserted between successive convolutional layers in a CNN architecture. Their function is to progressively reduce the spatial size of the representation, to reduce the number of parameters and computation in the network providing translation invariance and dimensionality reduction, leading to less computational requirements and preventing overfitting [11].

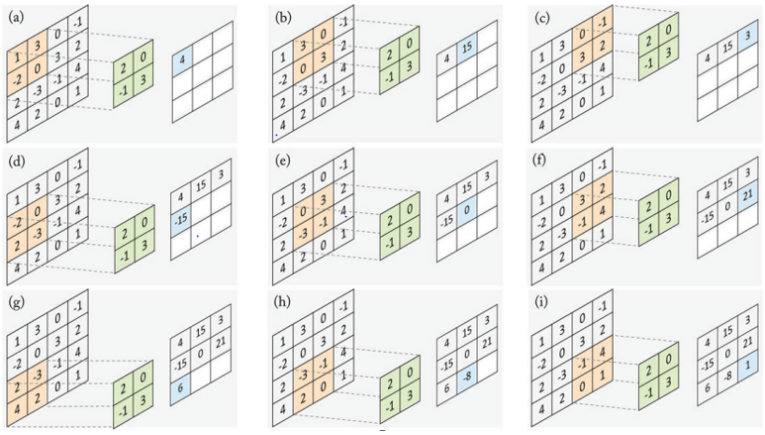


Figure 2.3: Convolutional Layer Architecture [11].

Max pooling is the most widely used pooling operation and works by defining a spatial neighborhood usually a 2x2 window and taking the maximum element from the rectified feature map within that window. It can also be understood as a "feature detector" that retains only the highest value in a particular feature map region, discarding all other information as shown in Figure 2.4 [11].

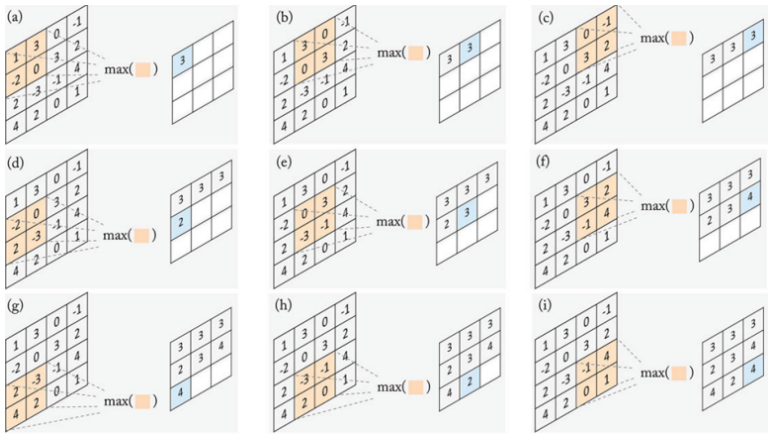


Figure 2.4: Max Pooling [11].

Furthermore, it is also possible to apply average pooling technique which calculates the average of the elements in a feature map region. Unlike max pooling which only keeps the maximum response, average pooling considers the average of the responses, thus retaining more information than max pooling. Historically, foundational deep learning architectures like LeNet-5 and AlexNet employed max pooling, and their successes cemented its use.

In the end of a CNN architecture it is used a fully connected layer. It corresponds essentially to convolution layers with filters of size 1x1. Each unit in a fully connected layer is densely connected to all the units of the previous layer. The purpose is to use these features for object detection, anomaly detection or even classifying the

input image into various classes based on the training dataset [11].

There are different types of CNN architectures and they use different amounts of layers, filter sizes, number sizes and the architecture choice is based on the nature of the problem.

2.1.4 Autoencoders

Autoencoders are neural network models trained to reconstruct their inputs using a smaller number of features. They can be useful for reducing the complexity of data and extracting features that capture more of the overall content of an image since they learn more abstract features [12].

In therms of architecture, the autoencoder is composed by 3 components: encoder, latent representation and decoder as can be seen in Figure 2.5.

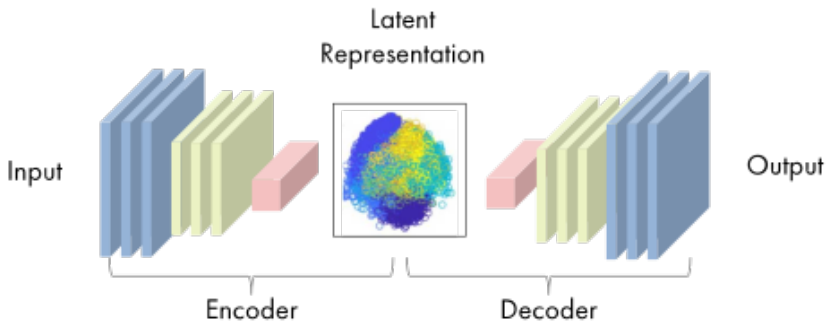


Figure 2.5: Autoencoder architecture [13].

The encoder contains a series of hidden layers responsible for compressing the input data into latent space by reducing the dimensionality of data. Neurons in the first encoding layer capture low-level features while deeper layers capture more complex features. The latent representation is the output of the encoder and consists in a set of essential features from input data. At the same time, decoder is trained to reconstruct the data as close to input as possible now based on the selected features (a smaller number of features) [14].

The Autoencoder is a machine learning based model for feature selection. This means it has the ability to capture both linear and non-linear relationships in the data due to their nonlinear activation functions. It is also an unsupervised model so it does not require a loss function and does not measure results against any pre-known ground truth (no need for labeling) [13].

When dealing with high amount of data, the autoencoder can be computationally expensive comparing with linear feature selection techniques like PCA and LDA.

2.2 Backbone Architectures in VAD

This section provides an analysis on SOTA backbones mostly utilized in VAD models so far. The term backbone refers to a feature-extracting network that processes input data into a certain feature representation, and its architecture may change depending on the type of approach used.

2.2.1 Review of Semi-Supervised approach

SS VAD models primarily rely on Residual Neural Network (ResNet) architectures, which introduce the concept of residual learning through shortcut connections that jump over some layers. This innovation addresses the vanishing gradient problem and enables the effective training of much deeper networks, which is crucial when only normal videos are used during training.

Residual learning, as illustrated in Figure 2.6, allows the network to learn the residual function $F(x) + x$, rather than directly mapping $F(x)$. This approach facilitates the backpropagation of gradients to earlier layers, thereby mitigating the vanishing gradient problem [15] [16].

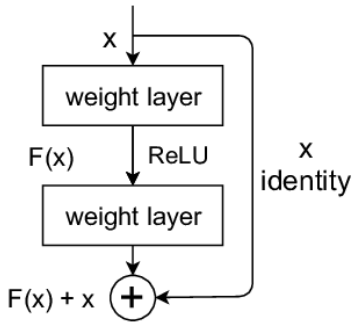


Figure 2.6: Residual Block. When the dimension of input x and output F(x) is the same, the shorcut connection is called Identity connection [17].

Traditional ResNets focused on increasing network depth, but this led to challenges such as the need for bottleneck blocks, as seen in Figure 2.7 b), to keep the network thin while increasing depth. WiderResNets were introduced as an evolution of this architecture, significantly increasing the network's width by multiplying the number of channels in each layer by a widening factor k. This approach not only improved training efficiency but also enhanced generalization on tasks like anomaly detection [18].

In contrast to deeper networks, WiderResNets maintain the basic ResNet block structure, as seen in Figure 2.7 a), while using wider convolutional layers, as Figure 2.7 c), which boosts the network's capacity to learn complex representations necessary for detecting anomalies. Moreover, the insertion of Wide-Dropout blocks in

the upper layers, Figure 2.7 d), has been shown to prevent overfitting and improve training by mitigating the issue of coadaptation [18].

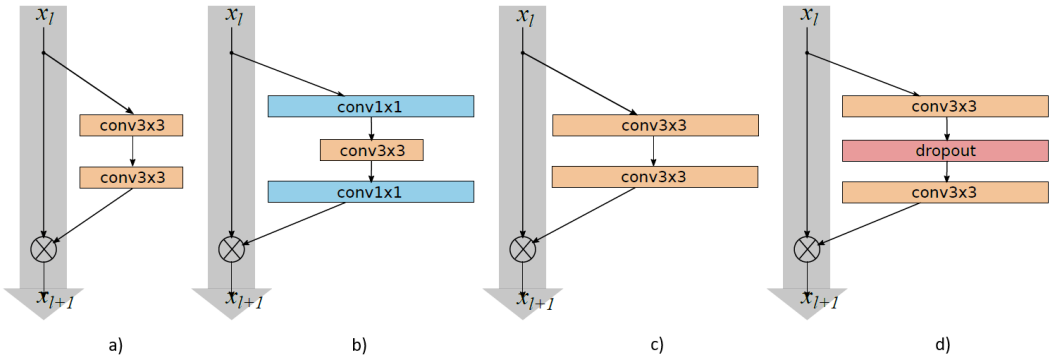


Figure 2.7: Residual Block Types: a) Basic; b) Bottleneck; c) Basic-Wide; d) Wide-Dropout. Batch normalization and ReLU precede each convolution (omitted for clarity) [18].

ResNet-based architectures have become an usual choice in SS approaches to VAD. Variants of these networks now often incorporate self-attention or spatial attention mechanisms to better focus on relevant features and suppress noise, further enhancing their effectiveness.

2.2.2 Review of Weakly-Supervised approach

Inception Networks, such as Inflated 3D (I3D), are pivotal in WS VAD models due to their ability to capture features at multiple spatial scales through parallel convolutions with different filter sizes (Figure 2.8). This design allows the network to efficiently recognize various features from both normal and anomaly videos, which is crucial during training of WS models [15] [19].

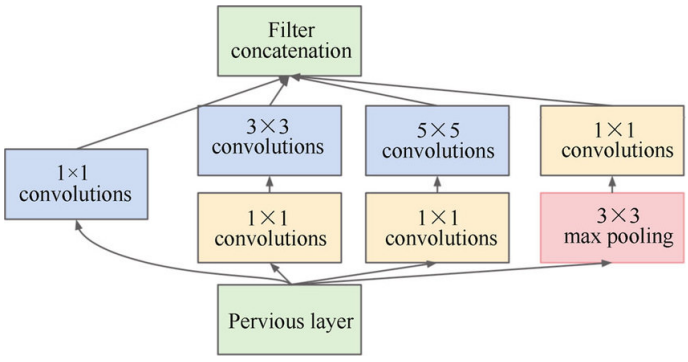


Figure 2.8: Example of an Inception Module [19].

The I3D model has particularly advanced this architecture by expanding 2D ConvNet filters and pooling kernels into 3D ($N \times N \times N$), enabling the seamless extraction of spatio-temporal features from video data (Figure 2.9). This adaptation

was introduced in February 2018 and marked a significant evolution from traditional 2D image classification models. Additionally, I3D employs a two-stream configuration (RGB and optical flow streams), further enhancing its ability to learn temporal patterns directly from video sequences [20].

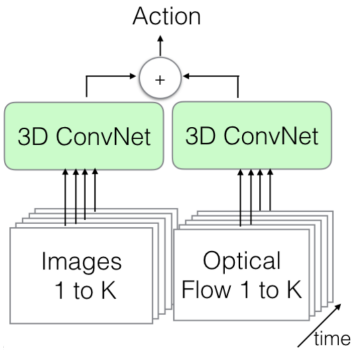


Figure 2.9: Two-Stream 3D-ConvNet Architecture [20].

Pre-training on the Kinetics Human Action Video dataset has substantially boosted I3D's performance, as shown in Table 2.1, particularly in action recognition tasks across the UCF-101 and HMDB-51 datasets. While the two-stream architecture generally performs better, the RGB-I3D stream alone is often preferred in VAD models due to its computational efficiency. Moreover, attention mechanisms have been increasingly utilized in VAD models to focus on human actors [20].

Model	UCF-101	HMDB-51
Two-Stream	88.0	59.4
IDT	86.4	61.7
Dynamic Image Networks + IDT	89.1	65.2
TDD + IDT	91.5	65.9
Two-Stream Fusion + IDT	93.5	69.2
Temporal Segment Networks	94.2	69.4
ST-ResNet + IDT	94.6	70.3
Deep Networks, Sports 1M pre-training	65.2	-
C3D one network, Sports 1M pre-training	82.3	-
C3D ensemble, Sports 1M pre-training	85.2	-
C3D ensemble + IDT, Sports 1M pre-training	90.1	-
RGB-I3D, Imagenet+Kinetics pre-training	95.6	74.8
Flow-I3D, Imagenet+Kinetics pre-training	96.7	77.1
Two-Stream I3D, Imagenet+Kinetics pre-training	**98.0**	80.7
RGB-I3D, Kinetics pre-training	95.1	74.3
Flow-I3D, Kinetics pre-training	96.5	77.3
Two-Stream I3D, Kinetics pre-training	97.8	**80.9**

Table 2.1: Comparison with state-of-the-art models on UCF-101 and HMDB-51 datasets, averaged over 3 splits. First set of rows contains results of models trained without labeled external data [20].

2.3 The Rise of Transformers and Self-Supervision

In recent years, the field of computer vision has seen a significant transition from traditional CNNs to Transformer-based architectures, particularly with the rise of ViTs. Furthermore, there is still an ongoing debate between self-supervised ans supervised techniques [21]. Two models leveraging SSL, CLIP and DinoV2, and one hybrid architecture model, NextViT, are presented.

2.3.1 Changes in Backbone Architectures

The dominance of ViTs in SSL has raised questions about the role of CNNs in the modern deep learning landscape. While Transformers have outperformed CNNs in many tasks, recent research suggests that the architectural limitations of older CNN models, such as ResNet, may have contributed to this performance gap. Studies have shown that by modernizing CNN architectures, as used in NextViT, which incorporates design elements inspired by Transformers, it is possible to close the performance gap. For instance, NextViT, with its larger kernel sizes and other architectural improvements, has been shown to perform competitively with Transformers in both supervised and SSL scenarios [21].

At the moment, CNN networks are only expected to be used in combination with Transformer architectures for better performance, leading to a possible new SOTA for CNNs in the era of Transformers. ResNets and Inception networks, which are still used in VAD models, are likely to be replaced by these new backbones [21].

In terms of architecture, ViT-based models have been increasingly adopted due to their integration of advanced mechanisms such as mask tokens and positional embeddings [22] [23]. Mask tokens challenge the model to predict missing patches of the image, creating a self-supervisory task that requires a holistic understanding beyond low-level image statistics. On the other hand, positional embeddings split the image into a series of patches, making the feature extraction process more efficient compared to traditional CNN networks [24].

2.3.2 Supervised Learning or not?

In recent years, the landscape of backbone models for computer vision tasks has expanded significantly beyond traditional supervised learning approaches. Historically, supervised learning, particularly using CNN pretrained on large datasets like ImageNet, was the default choice. However, the rise of SSL has challenged the dominance of supervised backbones by proposing alternatives that do not rely on labeled data for pretraining [25].

In Visual SSL, Contrastive Learning is a key approach, and is able to achieve top-tier performance by focusing on augmentation-invariant representations. It does this by bringing representations of different augmented views of the same image (positive

pairs) closer together while pushing apart those from different images (negative pairs). This method requires comparing features from many images simultaneously and often needs large batches of data or memory banks to gather sufficient negative pairs. DinoV2 uses a self-distillation-based loss and CLIP uses contrastive loss for this purpose [21]. SSL has achieved better adversarial robustness than supervised pretraining, so SSL models are less vulnerable to false alarms regarding VAD tasks [25].

In recent research, it was observed that Mixed-Architecture networks, like ConvNext, pretrained in a supervised manner, still outperform SSL models on many tasks. However, in scenarios where both SSL and supervised models were trained on similarly sized datasets, SSL models can outperform supervised ones. SSL techniques require larger batches or memory banks to work well, therefore the gap between these two approaches is narrowing as pretraining datasets continue to evolve [25].

SSL techniques have currently obtained better performance under ViT architectures, which are more sensitive to the amount of pretraining data and the number of parameters compared to CNNs. This suggest a huge margin of progression regarding ViT architectures on SSL [25].

2.3.3 CLIP

The CLIP model uses the technique of Contrastive Learning to map pairs of text and images instead of simply pre-training a model on an image classification dataset. For that, CLIP trains two different kinds of transformers: a Visual Transformer and a Text Transformer, which are more computationally efficient than CNNs, in order to encode a text descriptions and images into a vector. In Figure 2.10 it is obtained a vector representation from a mini-batch of images ($I_1 ... I_N$) and texts ($T_1 ... T_N$) through an image and text encoder, respectively, where the first image matches with the first text and so on [26].

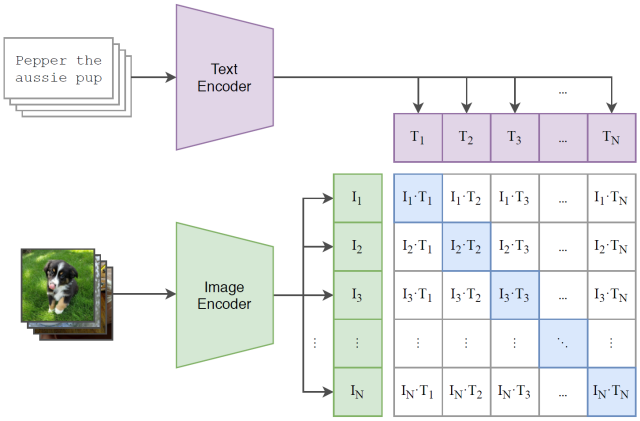


Figure 2.10: Contrastive pre-training. CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples [26].

For a given image representation, the model calculates the cosine similarity between all text representations in order to discover the most appropriate one to a particular image (contrastive objective). The larger mini-batches of images the more detailed representations it is possible to capture. In this case, the CLIP was trained with 400 million (image, text) pairs collected from the Internet while ImageNet, a widely used dataset for image classification, has 1.2 million.

The approach used surpasses the limitations of pre-training a model for image classification tasks because all images are lumped to a certain category in classification datasets, so there's no need to differentiate the individual images from each other. This makes the differences between images in the same category irrelevant since it doesn't concern the immediate classification problem.

To use this model as an image classifier, the image goes through the image encoder (a Visual Transformer) and then you see how similar is the text category with that image, obtaining its likelihood. By doing this, it is built a Zero-Shot Classifier without needing to train this classifier to a dataset. The Figure 2.11 shows this process [26].

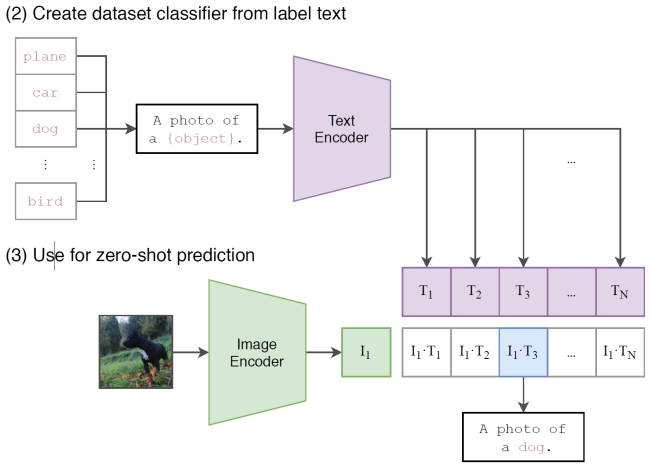


Figure 2.11: Zero-shot classifier. The learned text encoder embedds the names or descriptions on the target dataset's classes in order to classify the given image [26].

In therms of robustness to perturbation, the Zero-Shot CLIP model was compared with ResNet101 model fully-trained on ImageNet dataset. It is seen that the Zero-Shot CLIP outperforms the ResNet-101 when fed with variations of ImageNet itself in Figure 2.12. The model pre-trained with ImageNet decreases its performance when exposed to harder datasets with same classes as ImageNet, showing difficulties in recognising more detailed features in objects. The Zero-Shot CLIP is robust enough to capture more complex features and still recognises the same object in different scenarios [26].

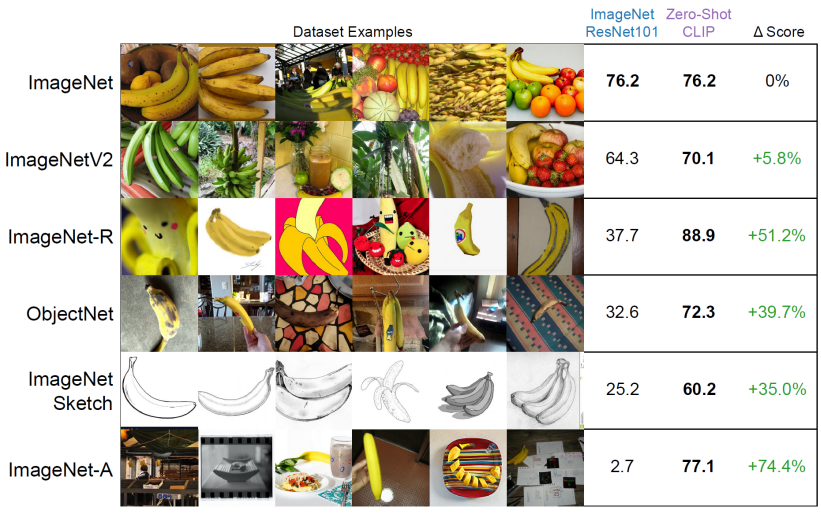


Figure 2.12: Zero-shot CLIP is much more robust to distribution shift than standard ImageNet pre-trained models. The performance of the best Zero-Shot CLIP Model, ViT-L/14@336px, is compared with a model that has the same performance on the ImageNet validation set, ResNet-101 [26].

For all these reasons, CLIP model is able to perform out of the box because it can be adapted to a widely variety of visual classification tasks without needing additional training examples. The accuracy of this classifier is often competitive with fully supervised models and it can extract more detailed features. The performance of Zero-Shot CLIP can be consulted in detail for various datasets in citation [26].

2.3.4 DinoV2

The DinoV2 is an image foundation model that is trained on broad data such that it can be applied across a wide range of use cases. For this purpose, a model generally uses segmentation masks or a text guidance (like CLIP) during training, however this type of approach is not able to detect every single action or emotion in a single image. For this reason, DinoV2 leverage images from small datasets and from internet and scales the number of images through an automated data generation pipeline as seen in Figure 2.13 [27].

The suggested pipeline aims to get quality training data at scale, and there are two sources of data: Uncurated and Curated. The Curated dataset combines pre-existing standard datasets used for different tasks and the Uncurated dataset consists of raw data collected from the internet through web crawling. After extracting image embeddings, the Deduplication step remove near-duplicate images from Uncurated data to incrase diversity among images. The last Retrieval step calculates the cosine similarity between Uncurated and Curated data, so images that are close to those from several Curated datasets are retrieved in a LVD-142M dataset, which ended up having 142M images and is used during training of this model.

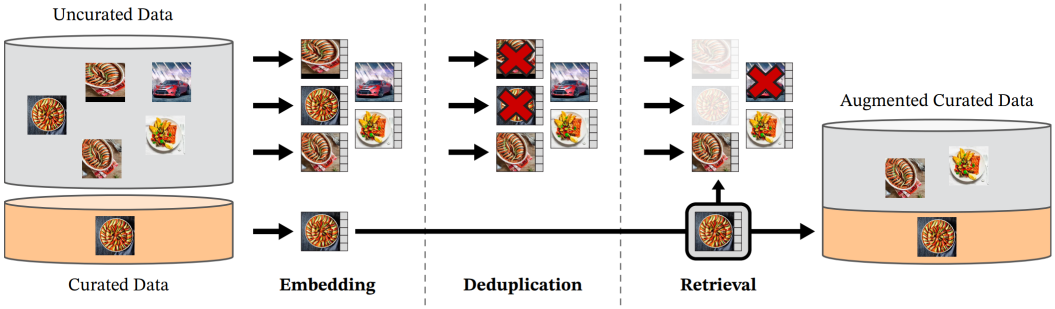


Figure 2.13: Overview of DinoV2 data processing pipeline. Images from curated and uncurated data sources are first mapped to embeddings. Uncurated images are then deduplicated before being matched to curated images. The resulting combination augments the initial dataset through a self-supervised retrieval system [27].

Dino stands for self-Distillation with No Labels, so it uses a discriminative self-supervised method to extract essential features directly from images, rather than relying on text descriptions. In this architecture, both features extracted from student and teacher networks are coming from the class token of a ViT, obtained from

different crops x_1 and x_2 of the same image x. Teacher contains a centering operation, which is updated through Exponential Moving Average and adds a biased constant every batch. This avoids the collapse to a constant function and this operation is followed by a softmax layer (consult [27] for more detail).

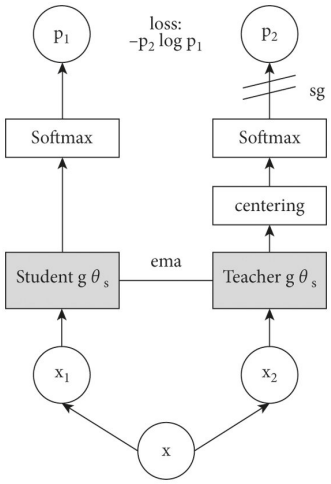


Figure 2.14: Architectural design of self-supervised DinoV2 model. SG indicates no reverse gradient propagation; and EMA indicates Exponential Moving Average update [27].

During operation, image crops that contains 50 % or more of the original image are fed into teacher network (global views), while image crops with smaller cutouts are fed into student network (local views). The teacher embedding of one of the global crops p_2 is compared with all embeddings from the student network p_1 and the Cross Entropy Loss is computed for each student embedding. The models seems to learn better when trying to match smaller parts of bigger object embedding [27].

DinoV2 has showed to perform effectively on many image tasks, including depth estimation, image classification and image segmentation. According to Meta Research, the performance of DinoV2 is "competitive or better than the performance of text-image models such as CLIP and OpenCLIP[1] on a wide array of tasks".

In Figure 2.15, DinoV2 features led to a much smoother depth estimation, and some objects, such as the cair on SUN RGB-D image are completely ignored by OpenCLIP[1]. The table 2.2 shows that DinoV2 model matches the accuracy of the OpenCLIP[1] features on image classification task [27].

Dinov2 leverages self-supervised learning techniques to allow for a model training process that does not require labels.

[1]OpenCLIP includes larger and independently trained CLIP models up to ViT-G/14, as seen in citation [28]

Feature	Arch	Food	C10	C100	SUN	Cars	Aircr	VOC	DTD	Pets	Cal101	Flowers	CUB	Avg
OpenCLIP	ViT-G/14	94.5	98.7	91.0	**84.0**	**96.1**	80.2	**89.3**	**86.0**	95.7	**98.1**	99.5	89.9	91.9
DINOv2	ViT-S/14	89.1	97.7	87.5	74.4	81.6	74.0	87.8	80.6	95.1	97.0	99.6	88.1	87.7
	ViT-B/14	92.8	98.7	91.3	77.3	88.2	79.4	88.2	83.3	96.2	96.1	99.6	89.6	90.1
	ViT-L/14	94.3	99.3	93.4	78.7	90.1	81.5	88.3	84.0	96.6	97.5	99.7	90.5	91.2
	ViT-g/14	**94.7**	**99.5**	**94.4**	78.7	91.4	**87.2**	89.0	84.5	**96.7**	97.6	**99.7**	**91.6**	**92.1**

Table 2.2: Accuracy comparison between OpenCLIP and DinoV2 feature models on 12 benchmarks covering objects, scenes and textures [27].

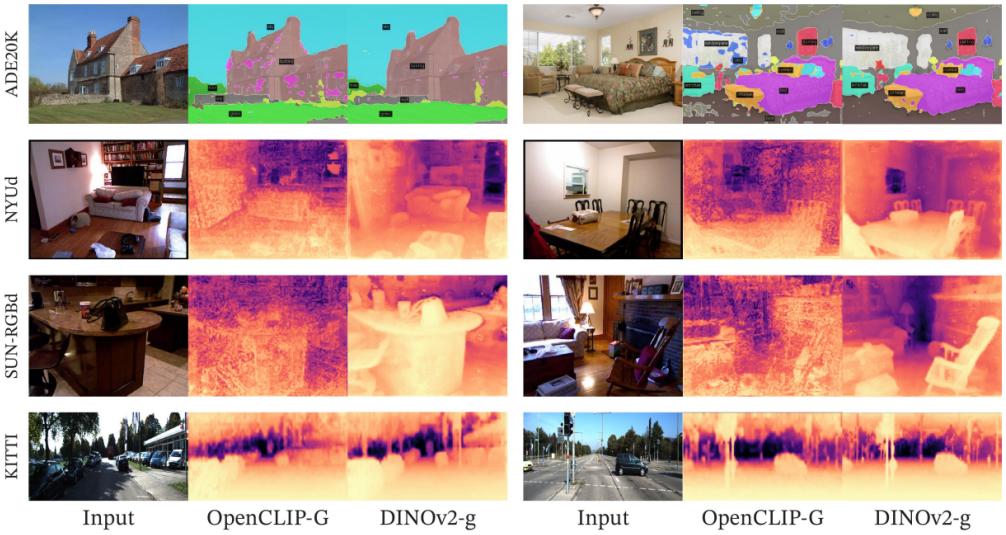


Figure 2.15: Segmentation and Depth Estimation comparison between OpenCLIP and DinoV2 model features on ADE20K, NYUd, SUN RGB-D and KITTI datasets [27].

2.3.5 NextViT

NextViT is a hybrid CNN-Transformer backbone designed to address the limitations of traditional ViTs and CNNs in real-world scenarios. While CNNs typically outperform ViTs in terms of inference speed, ViTs offer better performance for complex visual tasks. However, their high latency, due to mechanisms like Multi-Head Self Attention (MHSA), makes them inefficient for many real-world applications. NextViT aims to combine the strengths of both CNNs and ViTs, providing a solution that achieves a favorable balance between latency and accuracy [29].

The architecture of NextViT is built around two key components: the Next Convolution Block (NCB) and the Next Transformer Block (NTB). NCB captures local dependencies in visual data using a novel deployment-friendly mechanism called Multi-Head Convolutional Attention (MHCA), which integrates convolution with efficient attention. On the other hand, NTB is responsible for capturing global dependencies using a lightweight and efficient Efficient Multi-Head Self Attention

(E-MHSA). Next-ViT follows the hierarchical pyramid architecture equipped with a patch embedding layer and a series of convolution or Transformer blocks in each stage, as shown in Figure 2.16 [29].

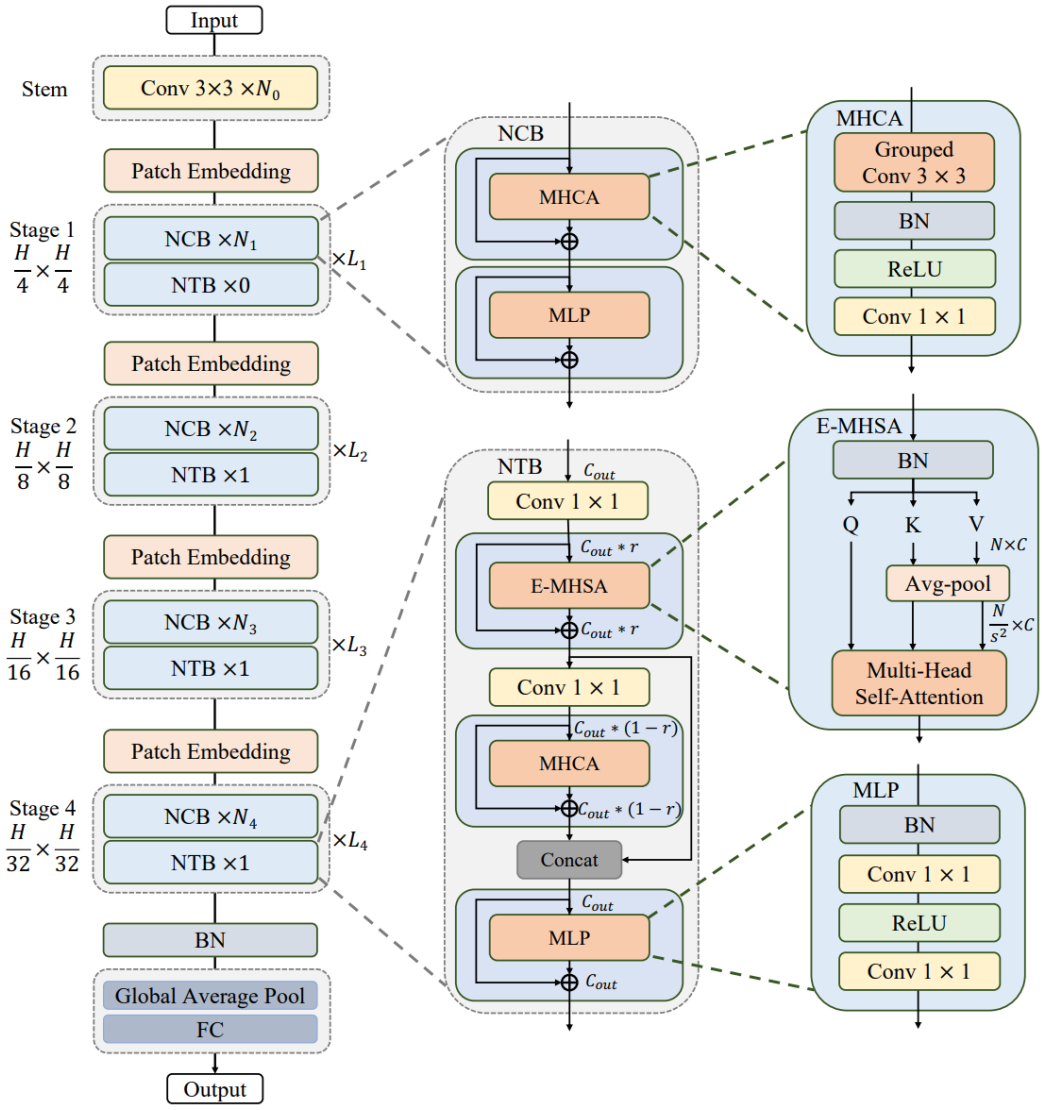


Figure 2.16: The left column is the overall hierarchical architecture of NextViT. The medium column show the NCB and NTB blocks. The right column are the detailed visualization of MHCA, E-MHSA and optimized MLP modules [29].

These blocks are integrated through the Next Hybrid Strategy, which stacks NCBs and NTBs in a novel hybrid arrangement. This architecture allows the network to leverage the local and global information processing strengths of CNNs and Transformers, respectively, while maintaining deployment efficiency across platforms like TensorRT and CoreML. Notably, NextViT exhibits significant improvements in tasks like object detection and semantic segmentation, surpassing previous hybrid architectures in terms of both accuracy and inference speed [29].

NextViT achieves a balanced trade-off between accuracy and latency, making it suitable for industrial applications that require real-time inference without sacrificing performance. It significantly outperforms traditional CNNs, like ResNet, in tasks such as object detection and semantic segmentation. For instance, NextViT achieves a 5.5-point improvement in mean Average Precision over ResNet on the COCO detection task, while maintaining comparable latency as Figure 2.17 suggests [29].

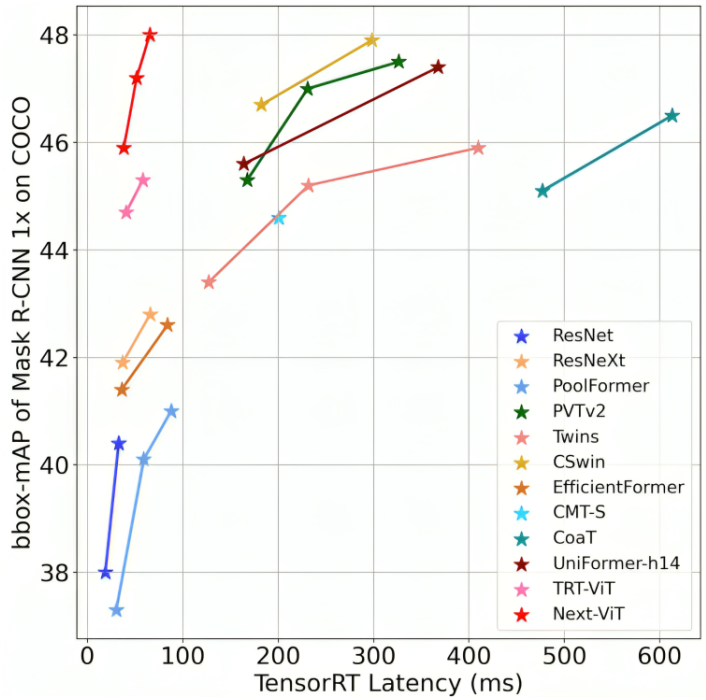


Figure 2.17: Comparison among Next-ViT and efficient Networks, in terms of accuracy-latency trade-off, when applied on COCO dataset in the task of detection [29].

As far as anomaly detection task is concerned, hybrid architectures like NextViT, which combine CNNs' local feature extraction with Transformers' global context modeling, may help in detecting both localized anomalies (like small objects or movements) and broader, scene-level anomalies.

Chapter 3

Methodology

This chapter describes the steps taken in the proposed benchmark. It presents the metrics utilized, the model selection process and a quick review on the chosen datasets.

3.1 Shifting to Emerging Backbones

This section provides the backbone tendency in VAD models. Furthermore, a recent benchmark that motivated the backbone choices for this dissertation is presented.

3.1.1 Analysis of VAD Backbones

The table 3.1 contains the feature extractors or backbones used in recent anomaly detection models, some of them with open-source code. From 2016, it can be seen an increase in research interest from SS to WS VAD approaches. During the last 2 years, there were launched more WS models than SS ones.

A notable aspect to consider is the wide array of feature extractors utilized in these models. They use sophisticated deep learning architectures such as Convolutional AutoEncoders, I3D, Convolutional 3D (C3D), and CLIP. The diversity of approaches underscores the intricate nature of the VAD task and emphasizes the necessity for tailored methodologies to address diverse scenarios.

In WS models, the I3D backbone, which leverages a CNN architecture called Inception, is widely used. However, a notably recent model like CLIP-TSA, published

in 2023, integrated advanced techniques from NLP like CLIP to improve scores, widening the usage of Transformers for anomaly detection.

In SS models, AutoEncoders and ResNet architectures, which is a type of CNN, have obtained the best results in this approach. Recent models have not proposed deep changes to these kind of backbones, with WiderResnet and ResNext keeping the essential structure, so there is no evidence of obtaining better results by utilizing other architectures, in contrast with WS models.

Model Name	**Backbone**	**Date**	**Supervision Type**	**Code Av.**
PEL4VAD [30]	I3D	January 2024		✓
TeD-SPAD [31]	U-Net[2] + I3D	August 2023		
CLIP-TSA [32]	CLIP	July 2023		✓
TEVAD [33]	I3D	June 2023		✓
CLAV [34]	I3D	June 2023	Weakly-Supervised	
CUPL [35]	I3D + VGGish	December 2022		
RTFM [36]	I3D	August 2021		✓
MIST [37]	I3D	April 2021		
Sultani *et al.* [38]	C3D	January 2018		
SLMPT [39]	U-Net[2]	October 2023		
USTN-DSC [40]	U-Net[2]	June 2023		
ASTNet [41]	WiderResnet	April 2022		✓
GCL [3]	ResNext	March 2022	Semi-Supervised	
AMC [42]	Conv-AE	August 2019		
FFP [43]	U-Net[2]	March 2018		✓
Hasan *et al.* [44]	Conv-AE	April 2016		

Table 3.1: List of the latest VAD models according to their backbones, the dates they were introduced, their supervision types, and code availability.

3.1.2 Backbones Benchmark

A recent benchmark evaluated a wide range of pretrained models across various computer vision tasks, including OOD classification, which is particularly relevant for anomaly detection. The benchmark focused on both conventional and emerging architectures, comparing their performance under different pretraining strategies.

[2]The U-Net is a special type of AutoEncoder used for feature extraction where the encoder and decoder parts are not separable. The ouput image depends directly on the input features, instead of only using the selected features from the latent representation. Consult [45] for better understanding of U-Net.

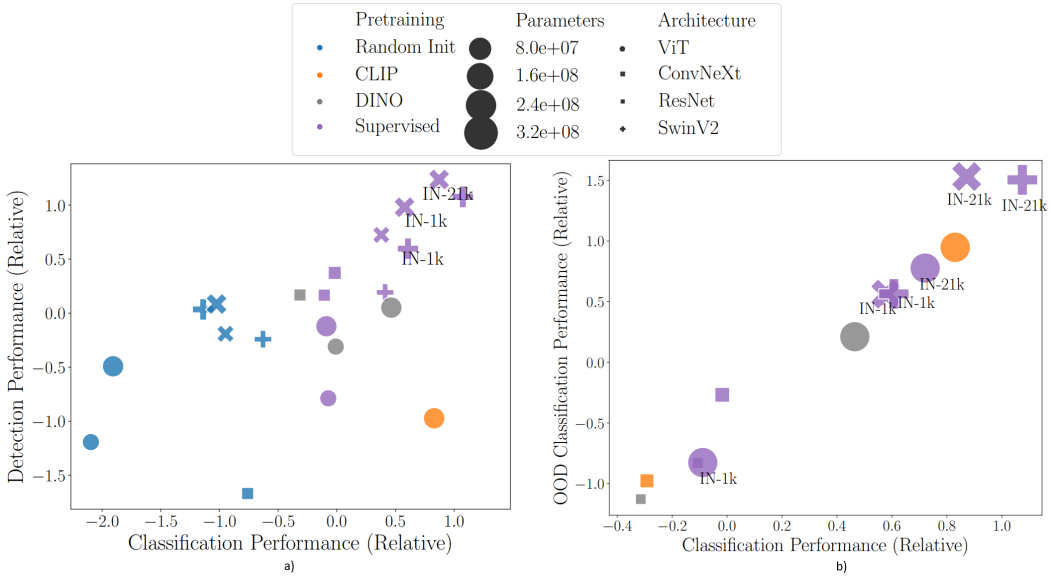


Figure 3.1: Performance for each model reported in terms of standard deviations above/below the mean averages across datasets: a) Comparison between classification and detection; b) Comparison between classification and OOD classification [25].

Figure 3.1 presents the performance correlated across tasks, Detection/Classification and OOD Classification/In-distribution Classification. Detection require backbones to extract features containing the precise locations of objects. OOD generalization is useful in real-world applications where models are often deployed on data which does not reflect their training set distribution [25].

The benchmark highlights the outstanding performance of backbones with hybrid architecture, such as SwinV2 and ConvNeXt, particularly when trained in a supervised manner on large datasets like ImageNet-21k. These backbones consistently outperformed others, including SSL backbones with ViT architectures, on tasks like OOD classification. Specifically, in OOD scenarios, SwinV2-Base and ConvNeXt-Base emerged as top performers, closely followed by self-supervised backbones like CLIP ViT-Base and Dino ViT-Small [25].

CLIP, which is a visual-language model, demonstrates strong OOD generalization capabilities and Dino model, which utilizes self-supervised learning, also shows competitive performance.

Based on the findings from this benchmark, the backbones selected for this study are presented in table 3.2. DinoV2 was published in February 2024, months after the publication of the presented benchmark (November 2023), so it has improved its performance on OOD and Detection tasks.

A recent research from July 2024 suggests that hybrid architectures trained in a supervised manner, such as ConvNext and NextViT, obtain higher robustness likely

because of pretrained ImageNet variants, on the other hand CLIP has a higher shape bias and makes less classification errors relative to their ImageNet accuracy [46].

Backbone	**Method**	**Pretrain Data**	**Params**	**Input-Size**
ViT-Large	OpenCLIP	ImageNet-12k WIT-400M	304.5M	336x336
ViT-Large	DinoV2	LVD-142M	304.4M	518x518
NextViT-Large	Supervised	Unknown-6M	57.9M	384x384

Table 3.2: Backbones chosen for this study.

3.2 Baseline Definition

In this section, the models of each approach used as baseline are identified. The architecture characteristics are then described while emphasizing key points.

3.2.1 Weakly-Supervised Models

WS models have been receiving more attention in recent years with the implementation of the encoder-agnostic method. This architecture leverages a vanilla feature encoder that receives extracted features from videos without being tailored to a particular task [4]. Table 3.3 list some details on the code shared by the authors of the models selected for the benchmark of this dissertation.

Published	Model	Availability	ML Framework
IEEE 2024	PEL4VAD	GitHub Repository [47]	PyTorch
ICIP 2023	CLIP-TSA	GitHub Repository [48]	PyTorch
CVPR 2023	TEVAD	GitHub Repository [49]	PyTorch
ICCV 2021	RTFM	GitHub Repository [50]	PyTorch

Table 3.3: Available source code of the surveyed WS models.

Multiple Instance Learning

The baseline models used in this approach utilize MIL technique, purposed for the first time by Sultani as Figure 3.2 illustrates. In MIL, an input untrimmed video is usually divided into short segments (called snippets), then a backbone is followed to extract feature embeddings and a prediction head generates an anomaly score in the end of the framework [38]. Thus, Weakly-Supervised models consist of two modules: video processing backbone and a prediction head.

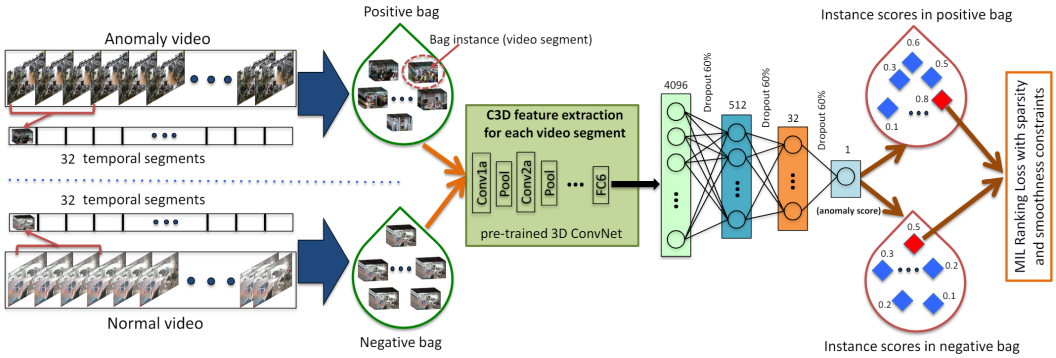


Figure 3.2: Multiple Instance Learning framework presented by Sultani [38].

The video processing backbone convert a video segment input into an embedding of a certain dimension and table 3.4 presents the feature dimensions accepted by each model. The majority of models incorporate a TSA mechanism in their architecture, which generates a reweighed attention feature by measuring the degree of abnormality of snippets.

Model	Backbone	Feature Dimension	Attention Mech.
PEL4VAD [47] (2024)	I3D	1024	✓
CLIP-TSA [48] (2023)	CLIP	512	✓
TEVAD [49] (2023)	I3D	1024/2048	-
RTFM [50] (2021)	I3D	1024/2048	✓

Table 3.4: Comparison of models' characteristics regarding their Backbone, Feature Dimension, and usage of Attention Mechanisms.

The prediction head, takes the feature embeddings and predict an anomaly score as output. Modern models leverage MLP-based networks with convolution layers to learn from temporal dependencies, RTFM and TEVAD also include Pyramid Dilated Convolutions (PDC) into this module.

This MIL framework treats each video as a bag containing several video segments: a bag is called positive if the input video contains an anomaly event, otherwise it is called a negative bag. These key ideas are utilized among this different model architectures.

During training, the models compute a ranking loss through positive and negative bags. The loss function compares the highest anomaly score from a positive bag with the highest score from a negative bag and the model is penalized if the maximum score from a negative bag is higher than from a positive bag. Apart from the top-1 ranking loss used in Sultani framework, models have been considering a different variation of ranking loss which consists in computing the mean value of the top-k anomaly scores, applied by all models in table 3.3 [38].

Mixing Text and Visual Modalities

Innovative approaches have been under experiments to improve VAD performance through multi-modality techniques. PEL module is a part of MLP-Network in PEL4VAD model that leverages anomaly class names such as 'fighting' or 'robbery' as keyword to query related words from a public semantic network called ConceptNet. The model is trained to align visual embeddings of video segments likely anomalous with text embeddings from anomaly class generated by CLIP, while pushing them away from the text embeddings of words related to 'normal' or 'non-violence', as shown in Figure 3.3 [30]. This technique reduces the False Alarm Rate (FAR) and allows to distinguish normal features inside anomalous video segments through align/repulse.

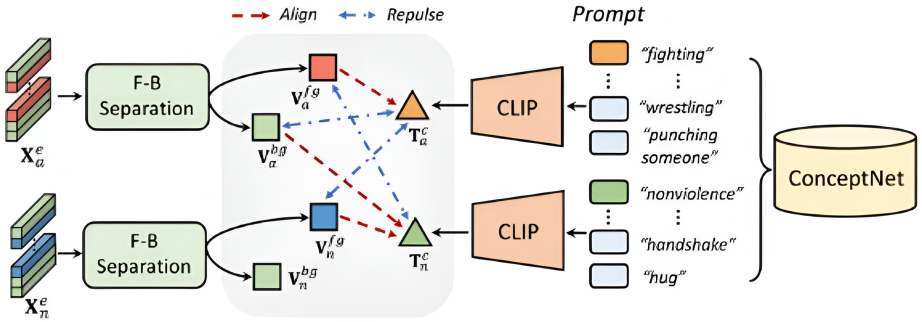


Figure 3.3: Prompt Enhanced-Learning module [30].

TEVAD model integrates both text and visual modalities to enhance video anomaly detection by generating dense captions for snippets of a video. Figure 3.4 shows the predicted anomaly score and the contributions of each word to the prediction. The use of captions provides explainability to the model: the illustrated video is classified anomalous due to the "skating" action [33].

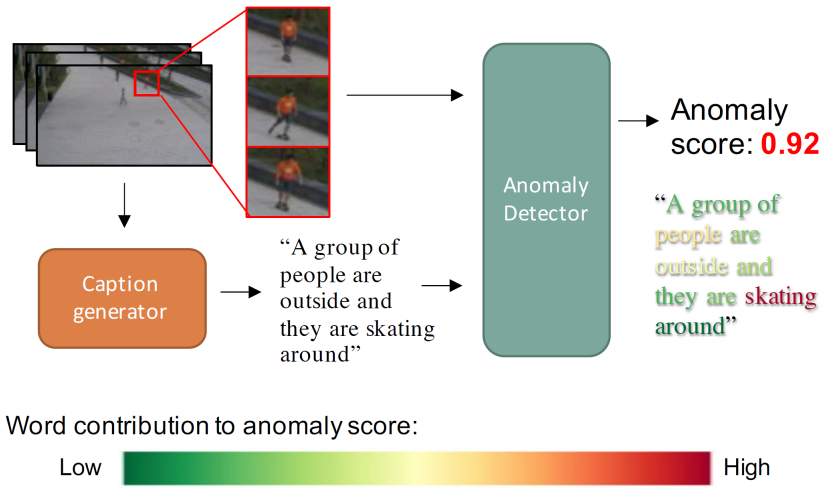


Figure 3.4: Improved video anomaly detection with captions [33].

3.2.2 Semi-Supervised Models

SS models fall in the category of One-Class Classification (OCC). This approach assumes that it is not possible to properly reconstruct or predict an abnormal event that has never been learnt, so the anomaly is detected because it deviates from the normality patterns previously trained. OCC is present in reconstruction-based and prediction-based SS methods [4]. Table 3.5 list some details on the code shared by the authors of models selected for this study.

Published	Model	Availability	ML Framework
CVPR 2018	FFP	GitHub Repository [51]	TensorFlow
AI 2022	ASTNet	GitHub Repository [52]	PyTorch

Table 3.5: Available source code of the surveyed semi-supervised models.

Regardless of the method used, whether reconstruction-based or prediction-based, SS approach employs two modules: a generator and a discriminator. The generator predicts or reconstructs the next video frame while the discriminator measures the quality of the predicted or reconstructed frame $\hat{I}$ compared to the ground truth frame I. So, the anomaly score comes from the calculation of Peak Signal-to-Noise Ratio (PSNR), using the formula 3.1.

$$\mathrm{PSNR}(I, \hat{I}) = 10 \log_{10} \left(\frac{\max(I)^2}{\frac{1}{N} \sum_{i=0}^{N} (I_i - \hat{I}_i)^2} \right) \tag{3.1}$$

The higher the PSNR, the smaller the difference between ground truth and predicted or reconstructed frames, which suggest a normal event, and vice-versa.

Prediction-based method

The presented baseline models use a prediction-based approach with the assumption that normal events are predictable, while abnormal events are not.

In general terms, the networks used for frame generation usually contain two modules: an encoder that extracts features by gradually reducing the spatial resolution and a decoder that gradually increases the spatial resolution to construct a frame.

The input data generated by the baseline models is presented in table 3.6. FFP uses RGB frames from a U-Net network, which is a type of Autoencoder, while ASTNet model uses extracted features with certain dimensions from a classic WiderResnet backbone to make predictions.

Both exploit both appearance and motion information in different ways. FFP introduces optical flow constraints enforcing the optical flow between predicted frames to be close to their optical flow ground truth (Figure 3.5), otherwise ASTNet employs

Model	Backbone	Input Type	Feature Dimension	Att. Mech.
FFP (2018)	U-Net	RGB	-	-
ASTNet (2022)	WiderResnet	Features	1024/2048	✓

Table 3.6: Comparison of models' characteristics regarding their Backbone, Feature Dimension, and usage of Attention Mechanisms.

a more robust 3D CNN network for learning both spatial and motion information, integrating attention mechanisms in its architecture as Figure 3.6 suggests.

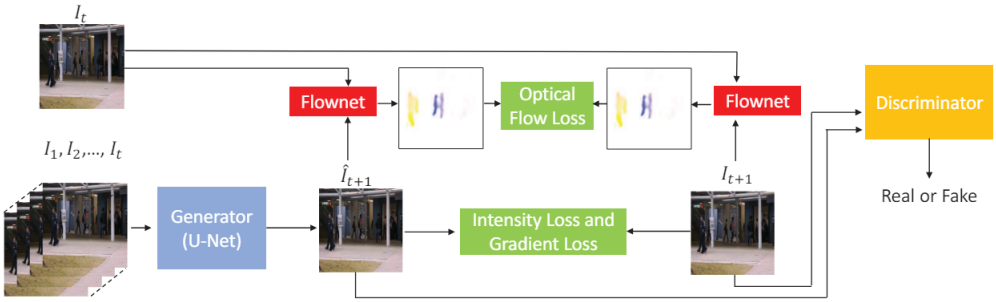


Figure 3.5: Overall Future Frame Prediction Framework structure [53].

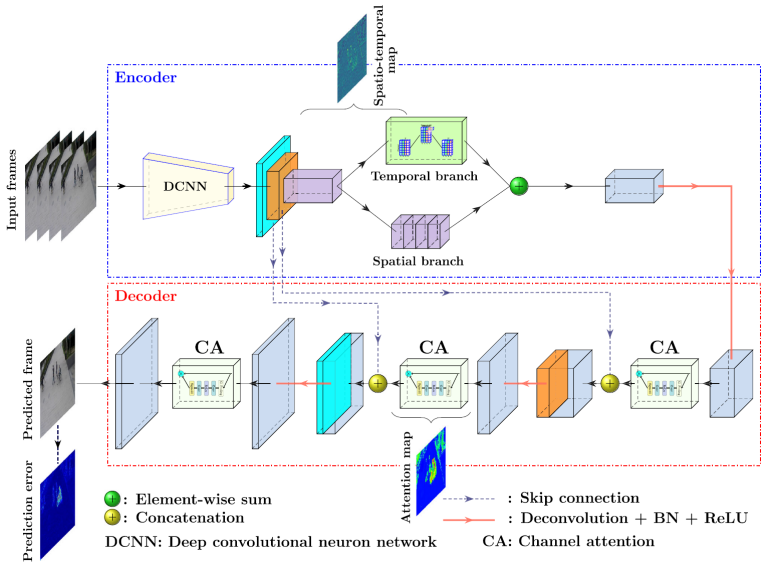


Figure 3.6: Structure of the ASTNet model [41].

Increasing Computational Efficiency

An extra computational power is required to run current SS models, so different techniques have been tried to overcome this issue. ASTNet model introduces a Temporal Shift on the architecture's temporal branch to achieve computational efficiency, shown in Figure 3.7 [41].

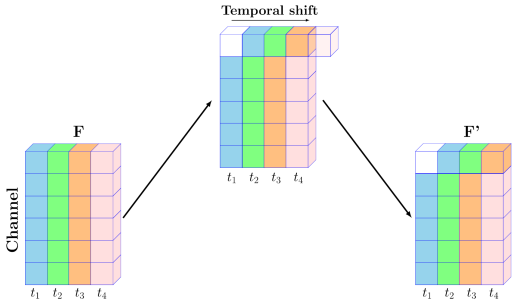


Figure 3.7: ASTNet's Temporal Shift technique [41].

3.3 Selection of Models for Benchmark

After a comprehensive search of models used across the two suggested approaches, WS and SS, models with open-source code are selected to reproduce the experiments and extend the research.

3.3.1 Weakly-Supervised Models

In WS learning, the four models were chosen for benchmarking: PEL4VAD, CLIP-TSA, TEVAD, and RTFM.

RTFM comes after the classic Sultani model and serves as the basis for subsequent models. The remaining models innovate in different perspectives regarding VAD problem: CLIP-TSA is the first VAD model to utilize a ViT backbone for feature extraction, while PEL4VAD and TEVAD mix visual and text modalities to add semantics to visual features and improve performance.

Firstly, the WS models are evaluated on UCF-Crime and trained on the corresponding dataset, as table 3.7 shows the results. CLIP-TSA has the highest ROC curve, showing a great disparity between TEVAD and RTFM, and surpasses PEL4VAD by 1.04 percentage points. However, PEL4VAD obtained higher precision, F1, and FAR scores. The significantly better F1-Score and FAR indicate a lower tendency for incorrect predictions, thus better overall performance.

Supervision	**Model**	**AUC**		**F1-score (%)**	**FAR (%)**
		ROC (%)	**P-R (%)**		
Weakly-Supervised	PEL4VAD* (2024)	86.76	**33.99**	**39.45**	**0.47**
	CLIP-TSA (2023)	**87.80**	33.12	31.70	32.37
	TEVAD (2023)	84.90	27.74	31.67	51.29
	RTFM (2021)	80.38	22.42	26.98	26.23

Table 3.7: Performance comparison of WS models on UCF-Crime Dataset.

Table 3.8 shows the results obtained by the models when evaluated on ShanghaiTech and trained on the corresponding dataset. Here, PEL4VAD obtained better

results across all metrics, proving to be the best performing model. The null occurrence of false alarms reflects the robustness of the model.

Supervision	Model	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
Weakly-Supervised	PEL4VAD* (2024)	**98.14**	**72.56**	**69.70**	**0.00**
	CLIP-TSA (2023)	97.40	60.32	59.72	3.68
	TEVAD (2023)	98.10	63.90	58.80	3.25
	RTFM (2021)	88.26	21.80	45.44	14.60

Table 3.8: Performance comparison of WS models on ShanghaiTech Dataset.

Based on this results, the best performing model in WS approach is PEL4VAD and the experiments proposed by this dissertation will focus in this model.

3.3.2 Semi-Supervised Models

In SS learning, fewer open-source models were available, so two models were chosen for benchmarking: FFP and ASTNet.

The FFP model is mentioned because it introduced the concept of using previous frames to compute a prediction of the following ones, leveraging RGB frames and optical flow, and serving as a basis for subsequent models. ASTNet uses the same philosophy but explores the appearance and motion information through a more complex network, while achieving computational efficiency.

Table 3.9 shows the results of the models when evaluated on the ShanghaiTech, UCSD Ped2, and Avenue datasets, trained on the corresponding dataset. The ASTNet model obtained a higher ROC and Precision-Recall (P-R) curve across all datasets compared to FFP.

Supervision	Dataset	Model	AUC	
			ROC (%)	P-R (%)
Semi-Supervised	UCSD Ped2	ASTNet* (2022)	**97.42**	**83.51**
		FFP (2018)	95.42	81.20
	ShanghaiTech	ASTNet* (2022)	**73.64**	**77.29**
		FFP (2018)	72.80	75.23
	Avenue	ASTNet* (2022)	**86.68**	**94.18**
		FFP (2018)	84.93	90.02

Table 3.9: Performance Comparison of SS models on ShanghaiTech, UCSD Ped2 and Avenue Datasets.

Based on this results, the best-performing model in the SS approach is ASTNet, and the experiments carried out in this dissertation will focus on this model.

3.4 Experimentation Strategy

This section presents how the experiments will be conducted, focusing on two different types of evaluation: Intra-Dataset Evaluation (IDE) and Cross-Dataset Evaluation (CDE). The main contributions of the dissertation are highlighted.

3.4.1 Intra-Dataset Evaluation

In IDE, the model is evaluated on the same dataset it was trained on, so the proposed benchmarks are shown in Table 3.10. The UCF and Shanghai datasets are targeted in this evaluation for being large-scale datasets.

PEL4VAD model is first applied with I3D backbone, then replaced by three emerging backbones: CLIP, DinoV2 and NextViT. The goal is to assess whether using features from SSL and hybrid architecture backbones offers any advantage when training and testing the model on the same dataset.

ASTNet is tested and trained on both the UCF and Shanghai datasets, with this dissertation introducing a new benchmark for UCF from scratch. The blue cells in Table 3.10 correspond to evaluations conducted by the authors of the models, while the green cells represent the contributions of this dissertation.

Model	Backbone	UCF	Shanghai
PEL4VAD (WS)	I3D SH	-	✓
	I3D UCF	✓	-
	C/D/N SH	-	✓
	C/D/N UCF	✓	-
ASTNet (SS)	WR SH	-	✓
	WR UCF	✓	-

Table 3.10: Benchmark for IDE in PEL4VAD and ASTNet models. The models are trained on UCF and Shanghai datasets and evaluated on the corresponding dataset. The blue cells represent benchmarks from the original authors, and the green cells represent benchmark contributions of this dissertation. Abbreviations: C/D/N = CLIP/DinoV2/NextViT; WR = WiderResnet.

3.4.2 Cross-Dataset Evaluation

In CDE, the model is evaluated on a different dataset it was trained on. For this purpose, the models are trained on large-scale datasets (UCF and Shanghai) and then evaluated on testing sets with different scenarios and features, as table 3.11 suggests.

Model	Backbone	Ped1	Ped2	Adoc	IITB
PEL4VAD (WS)	I3D SH	✓	✓	✓	✓
	I3D UCF	✓	✓	✓	✓
	C/D/N SH	✓	✓	✓	✓
	C/D/N UCF	✓	✓	✓	✓
ASTNet (SS)	WR SH	✓	✓	✓	✓
	WR UCF	✓	✓	✓	✓

Table 3.11: Benchmark for CDE in PEL4VAD and ASTNet models. The models are trained on UCF and Shanghai datasets and evaluated on Ped1, Ped2, Adoc and IITB Testing Sets. The blue cells represent benchmarks from the original authors and the green cells represent benchmark contributions of this dissertation. Abbreviations: C/D/N = CLIP/DinoV2/NextViT; WR = WiderResnet.

The dissertation applied testing sets from UCSD Ped1, UCSD Ped2, Adoc and IITB-Corridor datasets. In PEL4VAD model, the features of these 4 testing sets are extracted with the suggested backbones: I3D, CLIP, DinoV2 and NextViT, and the ASTNet used the original WiderResnet. The ASTNet authors have also built their own UCSD Ped2 Testing Set.

This evaluation falls within the task of OOD generalization, where the models' robustness and adaptability to new and unseen data are assessed. Large-scale datasets provide a diverse range of anomalous events, making them ideal for CDE.

3.4.3 Methods for Threshold Selection

During CDE, it is crucial to evaluate the performance of an anomaly detection model at a specific threshold, so the metrics Precision, Recall, Accuracy, F1 and FAR are calculated for each threshold. Two methods were employed to find the ideal threshold for anomaly detection:

- Method 1: This method maximizes the difference between the true positive rate and the false positive rate through the Youden's J Statistic (equation 3.2) and therefore generates the highest F1-score. It determines the closest point to the top-left corner of the ROC space, as exemplified in Figure 3.8;
- Method 2: This method minimizes the false alarm rate, which is the proportion of normal instances incorrectly classified as anomalies. The highest anomaly score in normal scenarios is identified, and a slightly higher threshold is chosen to ensure that most normal instances are not misclassified as anomalies.

$$J = \max(TPR - FPR) \tag{3.2}$$

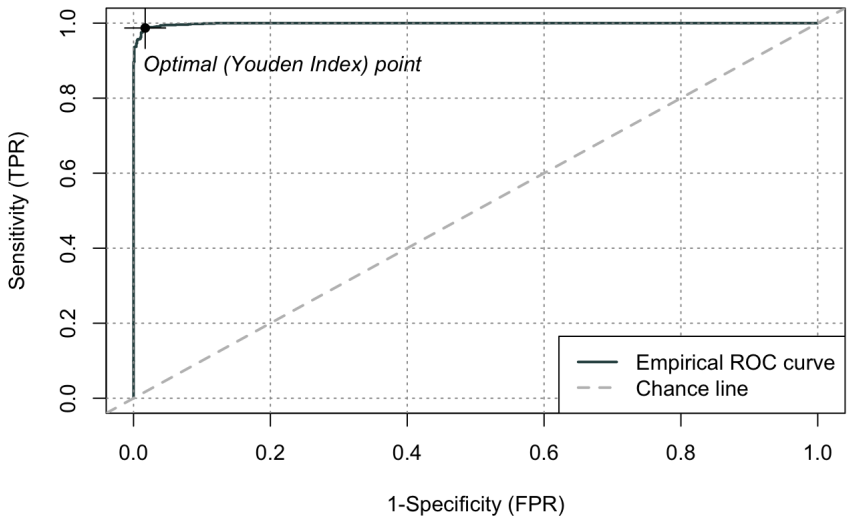


Figure 3.8: Example of ROC curve with Youden Index (optimal threshold point) [54].

The choice of the threshold depend on the trade-off you are willing to pay: the method 1 increases the ability to identify true anomalies sacrificing a higher false alarm rate; method 2 increases the ability to correctly identify true positives, but the model may fail to detect some anomalies. The final choice of the method may depend on the application requirements.

Chapter 4

Experimental Setup

In this chapter, the two large scale datasets leveraged in this experiment are described, relations between datasets and the composition of testing sets are also presented. The metrics served as basis for benchmark are explained.

4.1 Characterization of Datasets

The target VAD datasets for this experiment are presented in Table 4.1. This selection is composed by 6 datasets: 4 with a single camera view, and 2 with multiple cameras. Adoc and IITB-Corridor are the most recent datasets with 1080p resolution, the rest of them have lower resolutions.

Dataset	**Year**	**Views**	**HRA**	**NHRA**	**Resolution**	**Videos**
UCF-Crime [55]	2018	NA	13	1	320x240	1900
Shanghaitech [53]	2017	13	13	-	856×480	437
UCSD Ped1 [56]	2010	1	5	-	238x158	70
UCSD Ped2 [56]	2010	1	5	-	238x158	28
Adoc [57]	2020	1	5	-	1920×1080	4
IITB-Corridor [58]	2020	1	10	-	1920×1080	358

Table 4.1: Summary of datasets included in the research. Abbreviations: HRA = Human-Related Anomalies; NHRA = Non-Human-Related Anomalies; NA = Not Available.

The presented datasets provide an analysis of backbone's performance throughout different resolutions and the types of anomalies explored are those related to human activities, such as fighting, stealing, and skateboarding.

4.1.1 Testing Sets for Intra-Dataset Evaluation

The testing sets are evaluated in 2 types of benchmarks: IDE and CDE. In IDE the models are applied on UCF and ShanghaiTech Testing Set, as they are trained on the same datasets.

The table 4.2 has the composition of the videos in UCF and Shanghai Testing Sets and 4.3 presents the frame distribution employed in IDE for each method analysed in this benchmark: SS for ASTNet and WS for PEL4VAD model. In Semi-Supervision, abnormal videos comprise the Testing Set while Weak-Supervision include both normal and abnormal videos, so normal frames dominate the testing sets. This dissertation has built a UCF-Crime Testing Set for the ASTNet SS model, highlighted in green cells. The blue cells represent the testing sets from ASTNet and PEL4VAD authors.

Video Type	**Weakly-Supervised**		**Semi-Supervised**	
	UCF	**SH**	**UCF**	**SH**
Abnormal	140	24	235	107
Normal	150	175	0	0
Total	290	199	235	107

Table 4.2: Video composition of testing sets for IDE under WS and SS settings. The blue cells represent sets from the original authors, while the green cells indicate the set built for this dissertation.

Benchmark	**Dataset**	**Semi-Supervised**		**Weakly-Supervised**	
		Norm. (%)	**Abn. (%)**	**Norm. (%)**	**Abn. (%)**
IDE	UCF-Crime	67.57	32.43	87.83	12.17
	Shanghaitech	57.52	42.48	94.38	5.62

Table 4.3: This shows the distribution of normal and abnormal frames in the testing sets used in IDE. The green cells represent the testing set built for this dissertation, while the blue cells indicate the sets from the original authors.

4.1.2 Testing Sets for Cross-Dataset Evaluation

The proposed testing sets built from UCSD Ped1, UCSD Ped2, Adoc and IITB-Corridor original datasets are used for CDE. The testing sets are created in a WS

manner by leveraging normal and abnormal videos (marked in green cells), while the ASTNet authors have made a UCSD Ped2 Testing Set in a SS manner only containing abnormal videos, which is highlithed in blue cells. The composition of the videos within the sets is presented in table 4.4 and the distribution of normal and abnormal frames is shown in table 4.5.

Video Type	**Weakly-Supervised**				**Semi-Supervised**
	Ped1	**Ped2**	**Adoc**	**IITB**	**Ped2**
Abnormal	10	6	5	3	12
Normal	10	12	3	2	0
Total	20	18	8	5	12

Table 4.4: Video composition of testing sets for CDE under WS and SS settings. The blue cells represent the testing set from the original authors, while the green cells indicate the sets built for this dissertation.

Benchmark	**Dataset**	**Semi-Supervised**		**Weakly-Supervised**	
		Norm. (%)	**Abn. (%)**	**Norm. (%)**	**Abn. (%)**
CDE	UCSD Ped1	-	-	67.07	32.93
	UCSD Ped2	18.01	81.99	75.09	24.91
	Adoc	-	-	90.02	9.98
	IITB-Corridor	-	-	67.10	32.90

Table 4.5: Distribution of normal and abnormal frames in the testing sets of the datasets used in CDE. The blue cells denote the testing set from the original authors, while the green cells indicate the sets built for this dissertation.

For CDE, a proportion of nearly 70/30 normal to abnormal frames is considered balanced, therefore Adoc is the most unbalanced Testing Set. Normal frames are included in abnormal videos, so there is a disparity between normal and abnormal frames.

4.2 Large-Scale Training Sets

An analysis of 2 dimensions for Shanghaitech and UCF-Crime Training Sets is done: one accounts number of occurrences and other categorizes the scenarios found.

4.2.1 Composition of Training Sets

During training process, the datasets created under SS methods only use regular videos, so the normal videos comprise the training set. On the other hand, WS approach includes both normal and abnormal videos for training. The table 4.6 shows the training sets' video composition used by ASTNet and PEL4VAD models, in its respective supervision approach. Sets in blue cells are from the models authors while green cells stand for sets built for this dissertation.

Video Type	**Weakly-Supervised**		**Semi-Supervised**	
	UCF	**SH**	**UCF**	**SH**
Abnormal	810	63	0	0
Normal	481	175	482	330
Total	1291	238	482	330

Table 4.6: Video composition of training sets under WS and SS settings. The blue cells represent the training sets from the original authors, while the green cells indicate the training set created for this dissertation.

The ASTNet model was trained on a training set entirely built for this dissertation in order to run the purposed benchmark.

4.2.2 Anomaly Occurrences

This subsection is dedicated to the PEL4VAD model and WS approach since this is the only approach which includes abnormal videos within the models' training process. So the anomaly occurrences in each training set occur throughout the 63 ShanghaiTech and 810 UCF-Crime videos previously identified in table 4.6.

The diagram of the Figure 4.1 shows the structure of ShanghaiTech dataset with the different types of anomaly found in the different scenarios. Shanghai can be divided in 3 main scenarios: Courtryard, Road and Establishment Entrance. The Courtyard scenario contains 8 different camera angles, Road has 3 different camera angles and Establishment Entrance has 2 different camera angles.

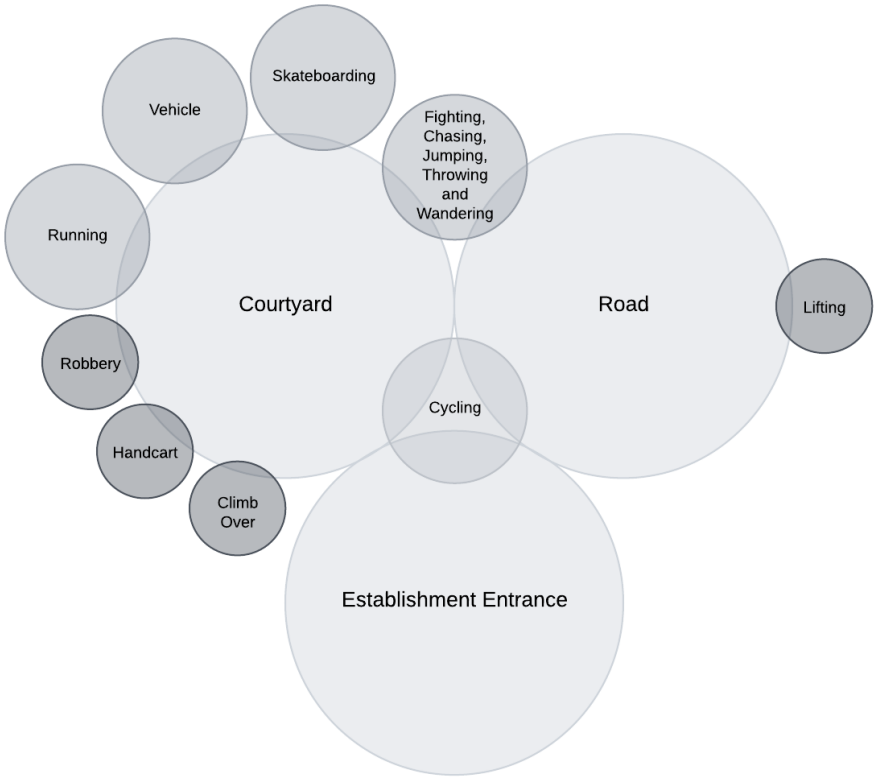


Figure 4.1: Structure of ShanghaiTech Training Set with the anomaly types found across the 3 main different scenarios: Courtyard, Road and Establishment Entrance.

In Shanghai 13 anomaly types were found as suggested in the Figure 4.1. The anomaly types with smaller and darker circle represent events that happened in 2 videos or less: Robbery, Handcart, Climb Over and Lifting. The most common anomalies found in this dataset are Cycling, followed by Running. Table 4.7 present the number of occurrences for each type of anomaly. The scenario that contains vaster types of anomalies is the Courtyard, predicting a greater performance of the model trained with this dataset when exposed to other Courtyard scenarios.

Type of Anomaly	**Number of Occurrences**
Cycling	27
Running	8
Skateboarding, Chasing, Jumping	6 each
Fighting	4
Vehicle	3
Throwing, Wandering, Robbery, Climb over	2 each
Handcart, Lifting	1 each

Table 4.7: Types of Anomalies and their number of occurrences on ShanghaiTech Training Set

The other large-scale dataset included in this study is UCF-Crime. This training set contains videos with random and specific scenarios such as public transports, stores, basketball playgrounds and football stadium stands. So it was not possible to present all the scenarios in a concise diagram as in ShanghaiTech 4.1. In fact, this atypical variation in UCF scenarios has been highlighted as a structural flaw because the models fail to learn patterns of normality, making normalcy as an open set [4]. The table 4.8 present the number of occurrences for each type of anomaly on UCF.

Type of Anomaly	**Number of Occurrences**
Robbery	145
Road Accidents	128
Stealing	95
Burglary	87
Abuse	48
Assault	47
Arrest, Fighting, Vandalism	45 each
Arson	41
Explosion	29
Shooting, Shoplifting	27 each

Table 4.8: Types of Anomalies and their number of occurrences on UCF-Crime Training Set

In UCF, the most common anomaly type is Robbery followed by Road Accidents. This allows to know the types of anomalies most easily captured by a model trained with this dataset.

4.2.3 Scene Categorization

During training, a model must learn how to operate in several conditions and scenarios. Applying models in scenarios similar to the training set may lead to a better generalization. In this case, Shanghaitech and UCF-Crime Training Sets are analyzed and categorized in 4 scenario types: Indoor or Outdoor environments in Crowded or Uncrowded places. A Crowded place is characterized by a high density of people while in Uncrowded places the interactions are sparse and human presence is low. A high variety of scenes, specially in UCF-Crime, motivated this type of categorization. Since this dissertation focuses on detecting human-related anomalies, the Crowded or Uncrowded categories were chosen for better assessment of the models as the higher or lower density of people could disturb the task of anomaly detection.

In Shanghai Training Set, 3 main scenarios are found as shown in Figure 4.2. All of them occur in the outdoor environment, whether crowded or not.

In SS approach, normal videos comprise the training set while in WS it includes all of the normal videos, but also a few anomalous videos. Both training sets contain around 35 % of scenes with high density of people (crowded), resulting in a majority of uncrowded scenarios, as shown in table 4.9.

Figure 4.2: Scenarios found in Shanghaitech Training Set: a) Courtyard; b) Road; c) Establishment Entrance.

Scenario Type	**Semi-Supervised (%)**	**Weakly-Supervised (%)**
Crowded (Outdoor)	35.37	35.19

Table 4.9: Distribution of scenario types on Shanghaitech Training Sets.

The fraction of crowded scenarios for certain types of anomalies among the 63 abnormal videos in the WS Shanghai Training Set (4.6) is presented in the table 4.10. The majority of cycling and skateboarding activities happen in low density places, vehicle and handcart anomalies only occur in uncrowded instances and the running activity is quite balanced regarding this type of categorization.

Scenario Type	**Cycling (%)**	**Skate. (%)**	**Vehicle (%)**	**Running (%)**	**Handcart (%)**
Crowded (Outdoor)	22.22	33.33	0.00	57.14	0.00

Table 4.10: Distribution of scenario types in videos with Cycling, Skateboarding, Vehicle, Running and Handcart anomalies for WS Shanghaitech Training Set.

In UCF-Crime, the proposed scenario categorization help to interpret the complexity of this dataset, Figure 4.3 show examples of this categorization.

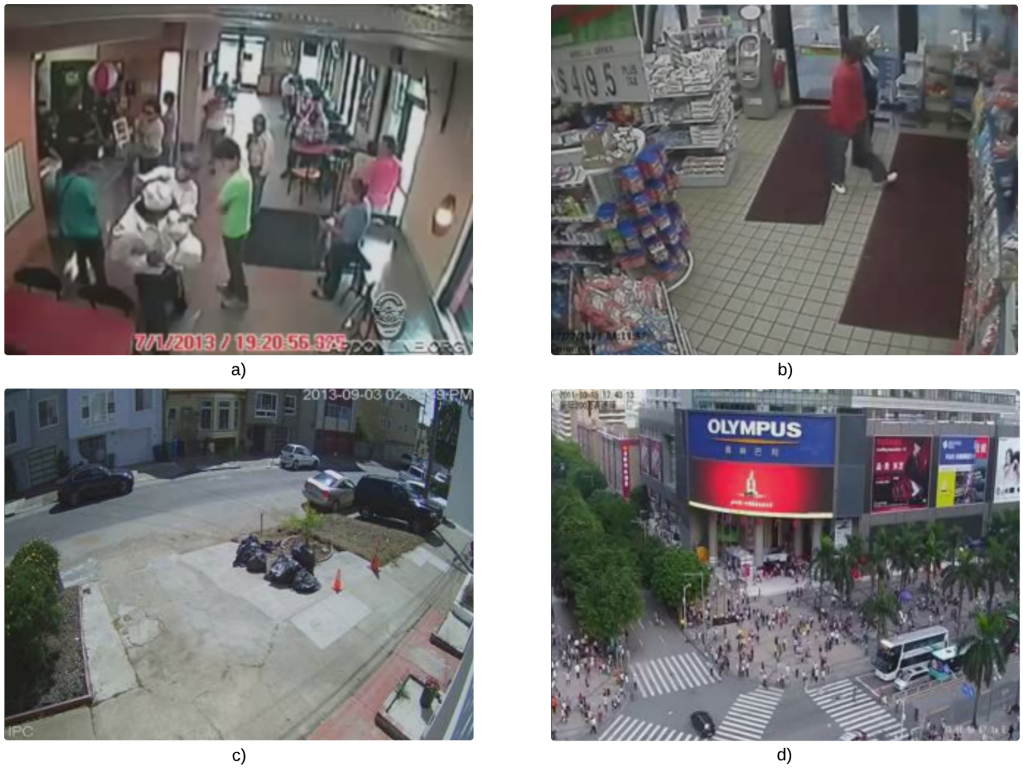


Figure 4.3: Example of scenario categorization on UCF-Crime Training Set: a) Crowded Indoor; b) Uncrowded Indoor; c) Uncrowded Outdoor; d) Crowded Outdoor.

The training sets used in both supervision approaches contain significantly fewer crowded videos, resulting in a majority of uncrowded scenarios. Regarding the environment, the WS Training Set is balanced with a nearly 50/50 proportion of indoor and outdoor scenarios, while the Semi-Supervised Training Set contains fewer outdoor scenarios, as shown in table 4.11.

Scenario Type	**Semi-Supervised (%)**	**Weakly-Supervised (%)**
Crowded (Out./In.)	26.56	27.76
Outdoor (Crow./Uncrow.)	37.64	47.91

Table 4.11: Distribution of scenario types on UCF-Crime Training Sets.

The fraction of crowded and outdoor videos with certain types of anomalies among the 810 videos in the WS UCF Training Set (4.6) can be seen in table 4.12. Abuse and assault anomalies occur mostly in uncrowded scenarios while fighting happen in crowded ones, nearly 70 % of the times. Outdoor scenarios happen less in abuse and fighting anomalies in contrast with assaults.

Scenario Type	Abuse (%)	Assault (%)	Fighting (%)
Crowded (In./Out.)	18.37	28.89	68.30
Outdoor (Crow./Uncrow.)	28.57	55.56	43.91

Table 4.12: Distribution of scenario types in videos with Abuse, Assault and Fighting anomalies for WS UCF-Crime Training Sets.

4.3 Relations between Datasets

After identifying the datasets, the relationships between them are denoted in order to understand whether the training conditions can impact the performance of the models when applied to more or less similar testing sets in later benchmark. Shanghai and UCF Training Sets are compared with other testing sets.

4.3.1 Anomaly Occurrences

This subsection aims to compare anomaly occurrences between Shanghai and UCF WS Training Sets and the proposed testing sets. On this count, a single anomaly video can contain multiple types of occurrences therefore each occurrence is not be equivalent to an anomalous video.

The table 4.13 presents the anomalies in common between Shanghai Training Set and the rest of testing sets, so a model trained with Shanghai is more likely to detect this kind of anomalies.

Type of Anomaly	Number of Occurrences				
	Ped1	Ped2	Adoc	IITB	SH
Cycling	7	5	3	0	27
Running	1	0	0	1	8
Skateboarding	2	2	2	0	6
Vehicle	2	1	0	0	3
Handcart	1	0	0	0	1

Table 4.13: Types of anomalies and their number of occurrences on Ped1, Ped2, Adoc, IITB Testing Sets and ShanghaiTech Training Set.

On the other hand, The only Testing Set which contains anomaly types in common with UCF-Crime is IITB, as shown in table 4.14. In the benchmark, a model trained on UCF Training Set would be able to detect any of the 3 occurrences shown in the table.

Type of Anomaly	Number of Occurrences	
	IITB	UCF
Abuse	1	48
Assault	1	47
Fighting	1	45

Table 4.14: Types of anomalies and their number of occurrences on IITB Testing Set and UCF Training Set.

4.3.2 Scene Categorization

The anomaly occurrences can happen in different scenario types. Since this work focuses on detecting human-related anomalies, the videos are categorized as crowded or uncrowded since higher or lower density of people could somehow affect the anomaly detection. Indoor and outdoor classification is also leveraged due to high scenario variations across UCF dataset, which may lead to further analysis.

All of selected testing sets happen in the outdoor environment and they have a single camera view, as shown in Figure 4.4. Ped1, Ped2 and Adoc datasets are pointed to a Courtyard while IITB points to a corridor.

Figure 4.4: The different scenarios among selected testing sets: a) UCSD Ped1; b) UCSD Ped2; c) Adoc; d) IITB-Corridor.

For now, it matters to analyse in which kind of circumstances each anomaly occurs. The table 4.15 compares the fraction of videos considered as crowded (so the rest is considered uncrowded) when certain anomalies occur between WS Shanghai Training Set (which contains anomalies) and the other testing sets. For example, the table shows that 85.71 % of cycling videos in UCSD Ped1 Testing Set happen in crowded places.

Anomaly Type	**Fraction of Crowded Videos (%)**				
	Ped1	**Ped2**	**Adoc**	**IITB**	**SH**
Cycling	85.71	60.00	33.33	-	22.22
Skateboarding	100	0.00	50.00	-	33.33
Vehicle	0.00	0.00	-	-	0.00
Running	0.00	-	-	100	57.14
Handcart	100	-	-	-	0.00

Table 4.15: Comparison of crowded scenario types between different testing sets (Ped1, Ped2, Adoc, IITB) and WS ShanghaiTech Training Set. The green and red cells represent more and less closer fractions to the Shanghai reference values, respectively.

The reference values of table 4.15 come from Shanghai Training Set. Testing sets with more than 30 % of fraction difference compared to reference values have their values painted in red, otherwise green.

The table 4.16 compares the fraction of crowded and outdoor scenarios between all testing sets and Shanghai Training Sets' videos. The proposed testing sets are very similar with the Shanghai Training Sets.

Scenario Type	**Testing Sets (%)**				**SH Training Set (%)**	
	Ped1	**Ped2**	**Adoc**	**IITB**	**SS**	**WS**
Crowded	45.00	33.33	37.50	0.00	35.37	35.19
Outdoor	100	100	100	100 [3]	100	100

Table 4.16: Scenario type comparison between Shanghai Training Sets and respective testing sets.

In UCF Training Set, anomalies can happen in crowded or uncrowded places within outdoor or indoor environments, therefore IITB Testing Set anomaly videos are compared with UCF regarding these 2 dimensions. In table 4.17, abuse and assault videos from IITB are more aligned with UCF Training Set, in contrast with fighting anomaly. Anomaly IITB videos all occur in the partial-outdoor scenario, while UCF is more divided as shown in table 4.18. Green cells represent a less

than 30 % of the fraction difference between the IITB and UCF reference values, otherwise it is red.

Anomaly Type	Fraction of Crowded Videos (%)	
	IITB	UCF
Abuse	0.00	18.37
Assault	0.00	28.89
Fighting	0.00	68.30

Table 4.17: Comparison of crowded videos between IITB Testing Set and WS UCF Training Set. The color green and red indicates more and less proximity of IITB values to the UCF reference values, respectively.

Anomaly Type	Fraction of Outdoor Videos (%)	
	IITB	UCF
Abuse	100 [3]	28.57
Assault	100 [3]	55.56
Fighting	100 [3]	43.91

Table 4.18: Comparison of outdoor videos between IITB Testing Set and WS UCF Training Set. The color green and red indicates more and less proximity of IITB values to the UCF reference values, respectively.

When comparing all of testing sets' videos with both UCF Training Sets, they are mostly composed by uncrowded scenarios. The table 4.19 show the fractions obtained.

Scenario Type	Testing Sets (%)				UCF Training Set (%)	
	Ped1	Ped2	Adoc	IITB	SS	WS
Crowded	45.00	33.33	37.50	0.00	26.56	27.76
Outdoor	100	100	100	100 [3]	37.64	47.91

Table 4.19: Scenario type comparison between UCF Training Sets and respective testing sets.

4.3.3 Resolutions and Comparison Summary

The resolutions of training sets can influence the performance of the model regarding the method and feature extractor used, so the table 4.20 shows the resolution transition from UCF and Shanghai Training Sets to the rest of the testing sets.

When training on Shanghai, the features are extracted from 856x480 image resolution, while UCF uses a 320x240 resolution. Transitioning to a bigger or lower image resolution can be compared throughout the different backbones in this benchmark.

Training Set->Testing Set	**Resolution Transition**
SH->Ped1/Ped2	856x480->238x158
SH->Adoc/IITB	856x480->1920x1080
UCF->Ped1/Ped2	320x240->238x158
UCF->Adoc/IITB	320x240->1920x1080

Table 4.20: Resolutions transition from UCF and SH Training Sets to the proposed testing sets.

The table 4.21 summarizes scenario similarities between training and testing sets on videos with anomalies in common. Adoc anomalies occur in similar crowded conditions as Shanghai, in contrast with the rest. Both Shanghai and other testing sets are exposed to the outdoor environment.

UCF Training Set is exposed to the indoor environment, while IITB is only exposed to a partial outdoor environment. In terms of crowd density, IITB is quite similar to this training set, as table 4.22 reveals.

Testing Set	**Similarities with Shanghai Training Set**	
	Crowd Density	**Environment**
Ped1	✗	✓
Ped2	✗	✓
Adoc	✓	✓
IITB	✗	✱[3]

Table 4.21: Comparison of the testing sets and their similarities with the ShanghaiTech Training Set: whether the place is crowded or not, outdoor or indoor environment.

Testing Set	**Similarities with UCF Training Set**	
	Crowd Density	**Environment**
IITB	✓	✱[3]

Table 4.22: Comparison of the testing sets and their similarities with the UCF Training Set: whether the place is crowded or not, outdoor or indoor environment.

[3]IITB scenario can be considered as partial outdoor since it is covered by a roof and contains open sides with visible greenery in the outside.

4.4 Metric Identification

The metrics used for evaluating VAD models heavily depend on the ROC curve, which plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold values. The Area Under Curve (AUC) of the ROC measures the model's ability to distinguish between normal and abnormal classes. An AUC score of 1 indicates perfect separation between classes, while an AUC score of 0 means the model predicts positive class as negative and vice-versa. An AUC score of 0.5 indicates no class separation capacity [59].

The TPR, also known as Recall, is the proportion of positive samples correctly identified by the model, whereas the FPR, also known as False Alarm Rate (FAR), is the proportion of negative samples incorrectly detected, ensuring the reliability of anomaly predictions. Equations 4.1 and 4.2 show the calculation of TPR and FPR rates, with TP, FN, and FP being True Positives, False Negatives, and False Positives, respectively [4].

$$\text{Recall} = \text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.1}$$

$$\text{FAR} = \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{4.2}$$

In addition to the ROC, the Average Precision (AP) has become a standard evaluation metric in VAD [30]. Precision is the ratio of True Positive predictions to the total number of positive predictions made by the model (including False Positives), as shown in equation 4.3. Average Precision is the average of the precision values calculated at each threshold, weighted by the change in Recall (TPR) from the previous threshold [4].

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.3}$$

The F1-score measures the harmonic mean of Precision and Recall (equation 4.4), integrating both into a single metric for a better understanding of model performance. This metric considers the types of errors (False Positive and False Negative) and not just the number of incorrect predictions, making it highly useful in areas like fraud prevention and other safety-critical applications [60].

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.4}$$

The ROC and AP curves measure model performance across all possible classification thresholds. In contrast, the F1-score relies on a specific threshold to evaluate the model, and its changes affect precision and recall. The proposed heuristic uses

the threshold that produces the highest F1-score in a Testing Set, creating a best-case scenario for direct comparison between models [4]. FAR also depends on a specific threshold; in this case, a positive rate with a threshold value of 50 % is employed as default for evaluation [30].

Chapter 5

Results

The chapter presents the results obtained throughout each evaluation. The ROC and P-R curves are generated during the Intra-Dataset Evaluation, while anomaly scores for each video frame are produced during the Cross-Dataset Evaluation.

5.1 Intra-Dataset Evaluation

In this section, the Intra-Dataset Evaluation results of PEL4VAD and ASTNet models are presented. This evaluation is a common approach for assessing the model's generalization capacity and it entails testing the models on the same dataset they were trained on.

5.1.1 Evaluation on Shanghaitech

Figures 5.1 and 5.2 show the ROC and P-R curves produced by the ASTNet and PEL4VAD models, respectively. Table 5.1 has the metrics produced by each model.

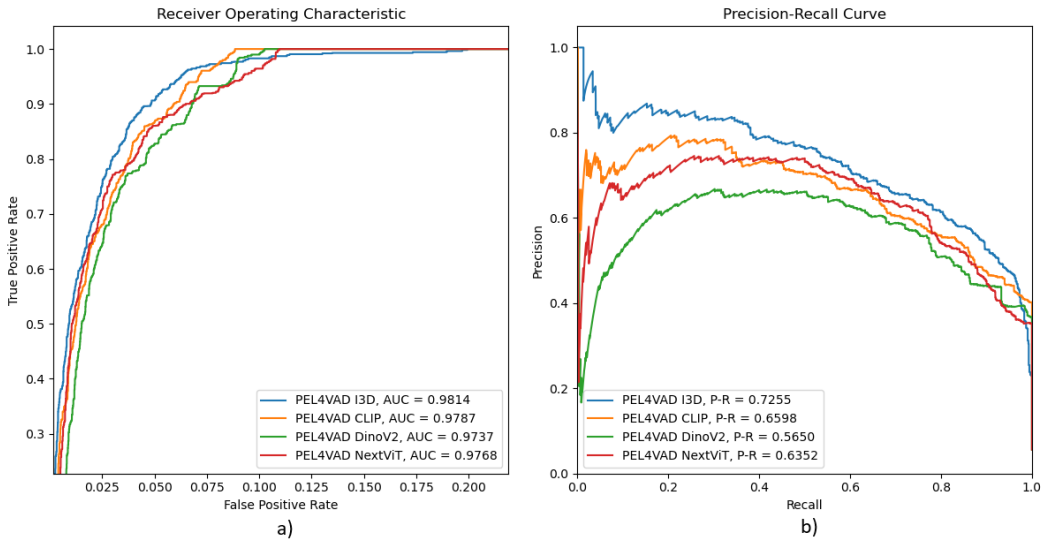


Figure 5.1: Curves of the proposed benchmark metrics obtained when testing the PEL4VAD model trained on Shanghaitech on the corresponding dataset: a) ROC Curves; b) Precision-Recall Curves.

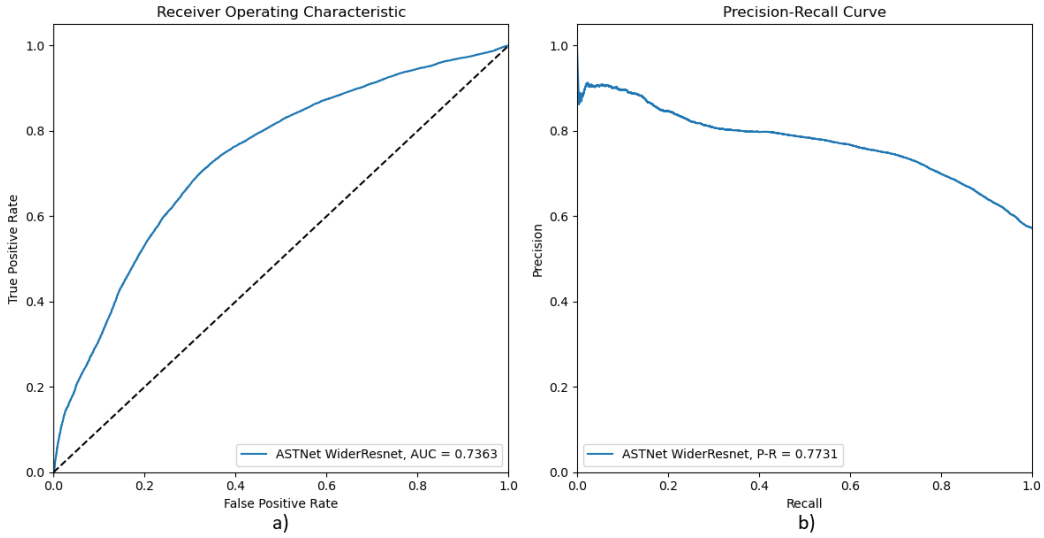


Figure 5.2: Curves of the proposed benchmark metrics obtained when testing the ASTNet model trained on Shanghaitech on the corresponding dataset: a) ROC Curves; b) Precision-Recall Curves.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT	97.68	63.52	67.77	0.95
	CLIP	97.87	65.98	66.33	0.29
	DinoV2	97.37	56.50	64.63	0.55
	I3D*	**98.14**	72.55	**69.70**	**0.00**
ASTNet (SS)	WiderResnet	73.63	**77.31**	60.00	5.13

Table 5.1: Comparison of the results achieved on Shanghai dataset for the benchmark of the models trained on the correspondent dataset using ASTNet and PEL4VAD models with different backbones.

5.1.2 Evaluation on UCF-Crime

This subsection is followed by the curves produced by each model PEL4VAD and ASTNet when evaluated on UCF-Crime, Figures 5.3 a) and 5.3 b), respectively. The table 5.2 has the metrics produced by the models.

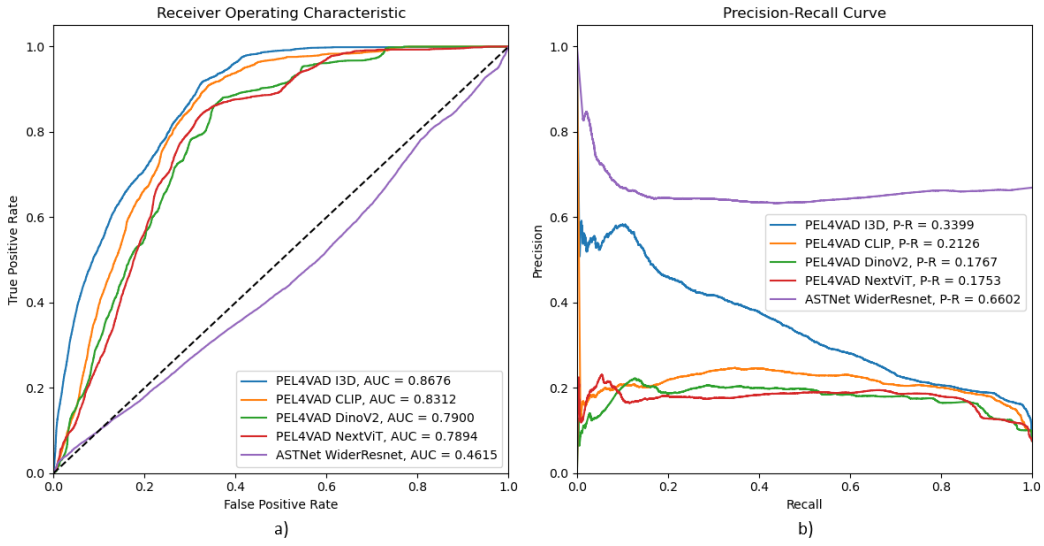


Figure 5.3: Curves of the proposed benchmark metrics obtained when testing the ASTNet and PEL4VAD models trained on UCF on the corresponding dataset: a) ROC Curves; b) Precision-Recall Curves.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT	78.95	17.55	30.20	2.63
	CLIP	83.12	21.26	33.25	**0.11**
	DinoV2	79.00	17.67	27.87	6.26
	I3D*	**86.36**	33.99	39.49	0.47
ASTNet (SS)	WiderResnet	46.15	**66.02**	**49.70**	26.23

Table 5.2: Comparison of the results achieved on UCF dataset for the benchmark of the models trained on the correspondent dataset using ASTNet and PEL4VAD models with different backbones.

5.2 Cross-Dataset Evaluation

In Cross-Dataset Evaluation the models are evaluated on a dataset they were not trained on. The intention is to analyse the performance of models trained on UCF and Shanghai datasets now applied in 4 different testing sets.

5.2.1 Evaluation on UCSD Ped1

The results applied on UCSD Ped1 are obtained when training the model on UCF and SH Training Sets. The metrics produced are seen in tables 5.3 and 5.4.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT UCF	**91.28**	**76.42**	**78.70**	15.54
	CLIP UCF	86.04	64.63	68.99	1.78
	DinoV2 UCF	81.19	61.03	71.76	**0.00**
	I3D UCF	55.53	34.30	51.50	77.14
ASTNet (SS)	WiderResnet UCF	59.79	74.18	50.30	31.66

Table 5.3: Comparison of the results achieved on UCSD Ped2 dataset for the benchmark of the models trained on UCF-Crime using ASTNet and PEL4VAD models with different backbones.

The Figure 5.4 shows the anomaly scores throughout the frames of the PEL4VAD CLIP model when trained on SH and UCF. The cycling activity lasts until frame 10000.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT SH	75.00	58.71	49.11	**0.00**
	CLIP SH*	**89.95**	**78.45**	**75.58**	0.59
	DinoV2 SH	42.73	31.86	49.11	**0.00**
	I3D SH	67.98	42.68	57.97	66.33
ASTNet (SS)	WiderResnet SH	59.62	74.99	50.30	30.70

Table 5.4: Comparison of the results achieved on UCSD Ped1 dataset for the benchmark of the models trained on ShanghaiTech using ASTNet and PEL4VAD models with different backbones.

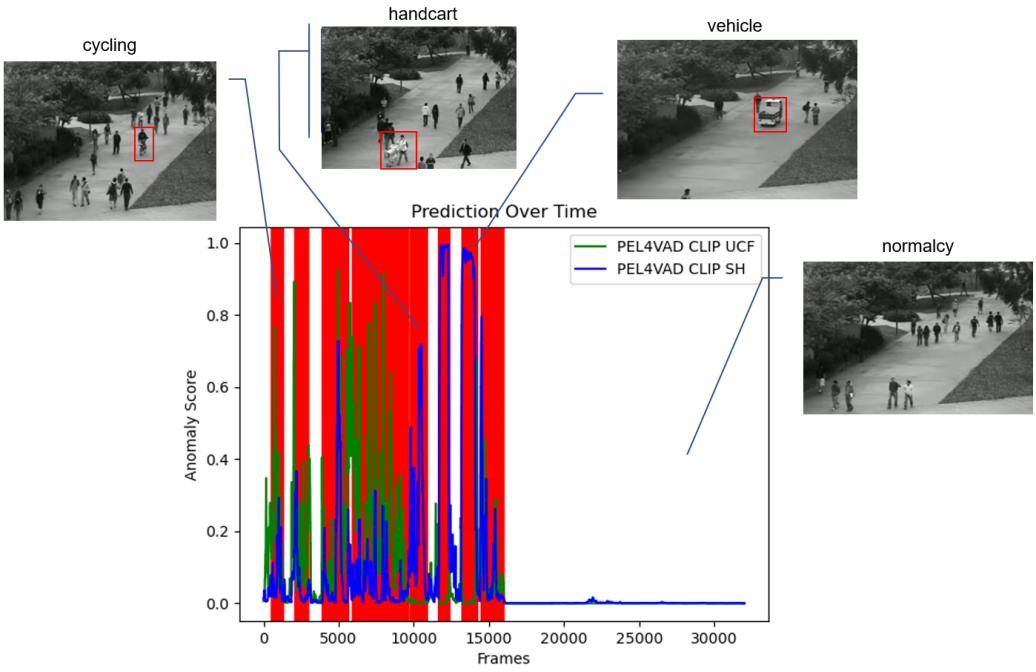


Figure 5.4: Anomaly scores produced by the PEL4VAD CLIP SH and PEL4VAD CLIP UCF models on UCSD Ped1 Testing Set built for this dissertation.

The Figure 5.5 shows the anomaly scores throughout the frames produced by PEL4VAD NextViT UCF model, which obtained great results.

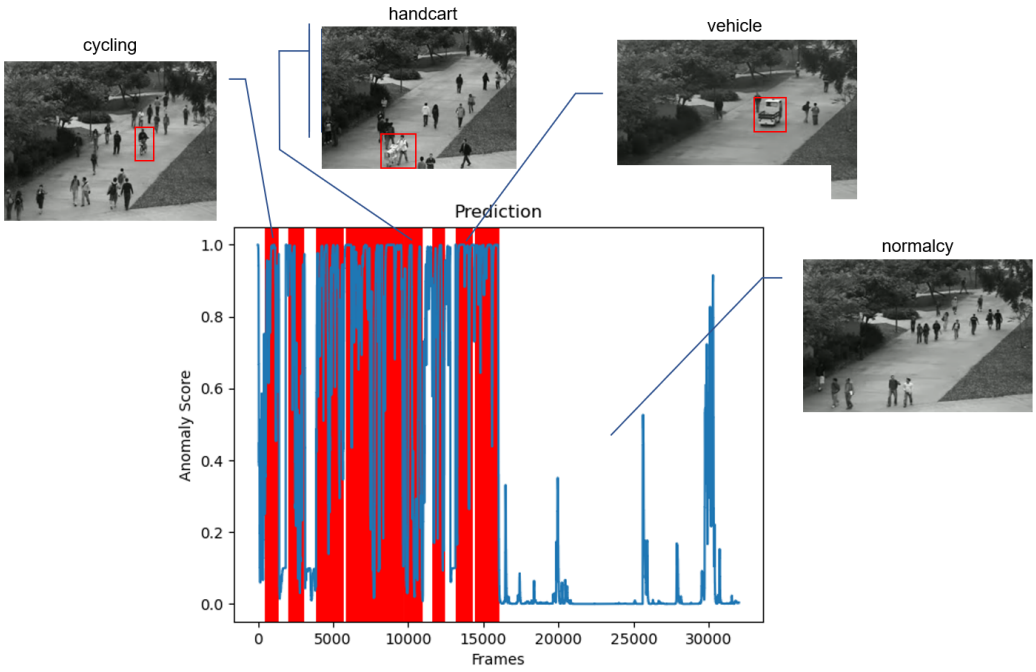


Figure 5.5: Anomaly scores produced by the PEL4VAD NextViT UCF model on UCSD Ped1 Testing Set built for this dissertation.

The table 5.5 shows the results obtained with the threshold from the proposed methods. The method 1 threshold misses less actual anomalies than method 2 threshold, but its less accurate in identifying true positives among all detected positives, hence increases the false alarms. Both methods can be chosen depending on the most worth trade-off.

Model	**Method**	**Prec.** (%)	**Rec.** (%)	**Acc.** (%)	**F1** (%)	**FAR** (%)
PEL4VAD CLIP SH	1*	63.14	**96.24**	80.49	76.25	27.11
	2*	69.15	74.50	80.88	71.73	**16.04**
PEL4VAD NextViT UCF	1*	66.76	95.62	83.08	**78.62**	22.98
	2*	**70.93**	86.02	**83.98**	77.75	17.01

Table 5.5: Comparison of different threshold values for the PEL4VAD model with CLIP backbone on UCSD Ped1 Testing Set.

The Figures 5.6, 5.7, 5.8 and 5.9 present 4 graphics in which the model classifies as anomaly the green shaded areas using the 2 suggested thresholds.

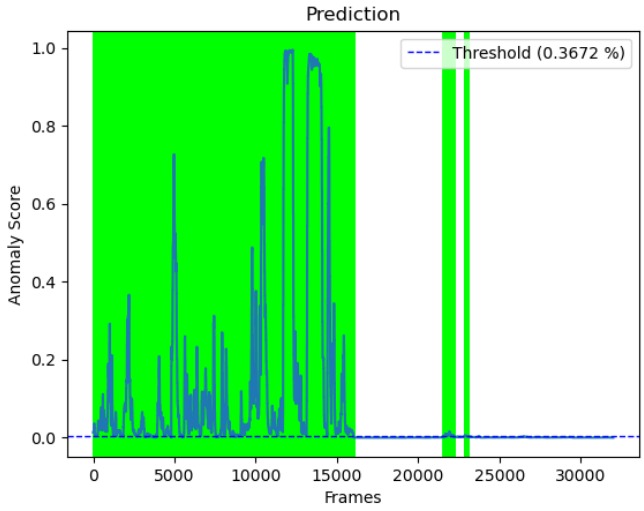


Figure 5.6: Anomaly detection produced by the PEL4VAD CLIP SH model on UCSD Ped1 Testing Set using method 1.

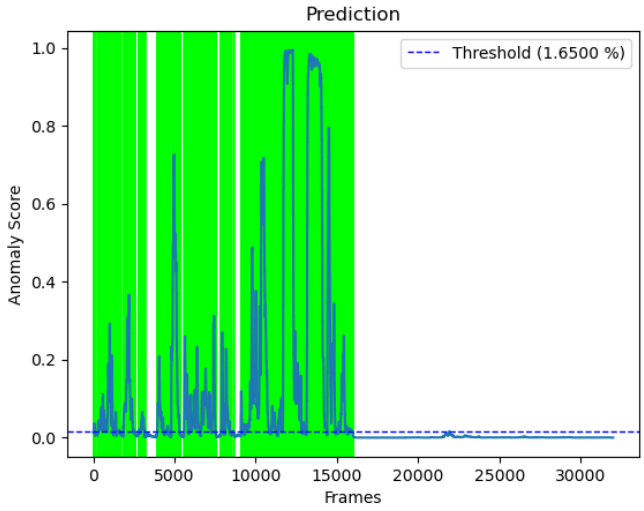


Figure 5.7: Anomaly detection produced by the PEL4VAD CLIP SH model on UCSD Ped1 Testing Set using method 2.

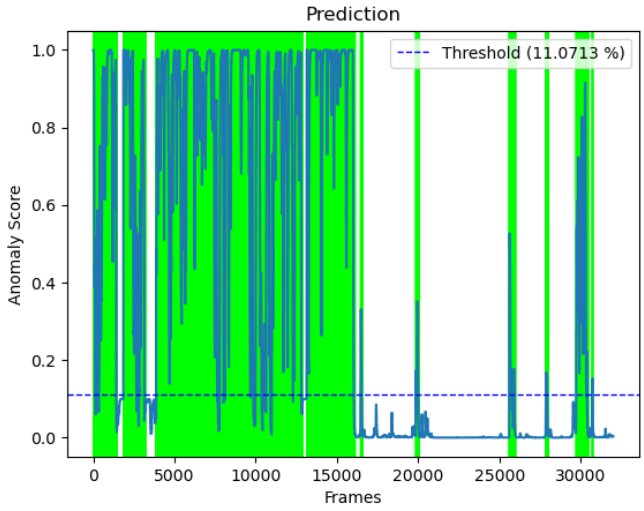


Figure 5.8: Anomaly detection produced by the PEL4VAD NextViT UCF model on UCSD Ped1 Testing Set using method 1.

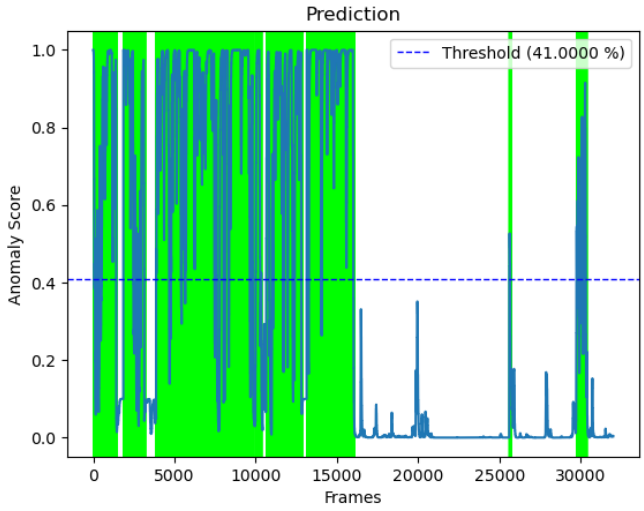


Figure 5.9: Anomaly detection produced by the PEL4VAD NextViT UCF model on UCSD Ped1 Testing Set using method 2.

5.2.2 Evaluation on UCSD Ped2

The results of the models when trained on UCF and SH datasets are presented on tables 5.6 and 5.7, respectively. For UCSD Ped2 Testing Set, ASTNet model is evaluated on the Ped2 Testing Set proposed by this dissertation and also on the Ped2 Testing Set built by ASTNet authors, which is made in a SS manner only with abnormal videos.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT UCF	56.31	29.14	46.05	74.31
	CLIP UCF	79.85	64.68	58.24	3.67
	DinoV2 UCF	32.66	18.20	39.37	62.45
	I3D UCF	65.82	58.76	53.73	4.46
ASTNet (SS)	WiderResnet UCF	64.34	**82.63**	42.76	30.77
	WiderResnet UCF (SS)	**89.56**	58.04	**90.45**	**2.63**

Table 5.6: Comparison of the results achieved on UCSD Ped2 dataset for the benchmark of the models trained on UCF-Crime using ASTNet and PEL4VAD models with different backbones.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT SH	84.07	73.46	39.11	**0.00**
	CLIP SH*	**95.11**	**86.07**	82.94	0.43
	DinoV2 SH	88.17	83.42	59.66	**0.00**
	I3D SH	86.72	70.25	39.11	100
ASTNet (SS)	WiderResnet SH	58.46	77.92	40.97	20.13
	WiderResnet SH (SS)	87.81	60.10	**90.45**	2.92

Table 5.7: Comparison of the results achieved on UCSD Ped2 dataset for the benchmark of the models trained on ShanghaiTech using ASTNet and PEL4VAD models with different backbones.

Figures 5.10 and 5.11 show the anomaly scores throughout the frames when applied on Ped2 Testing Set built by ASTNet authors and on the one proposed by this dissertation. ASTNet produced lots of noise in both testing sets and failed to detect normalcy.

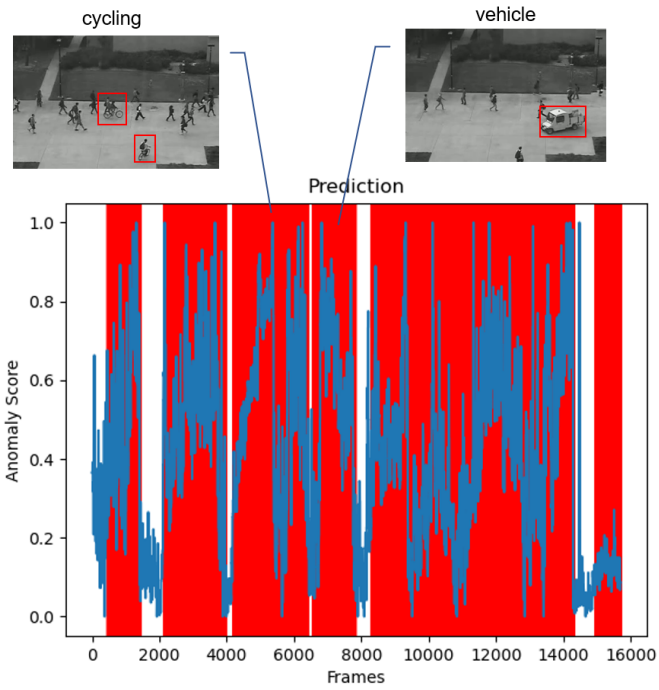


Figure 5.10: Anomaly scores produced by the ASTNet model on the original authors' UCSD Ped2 Testing Set (More Anomalies).

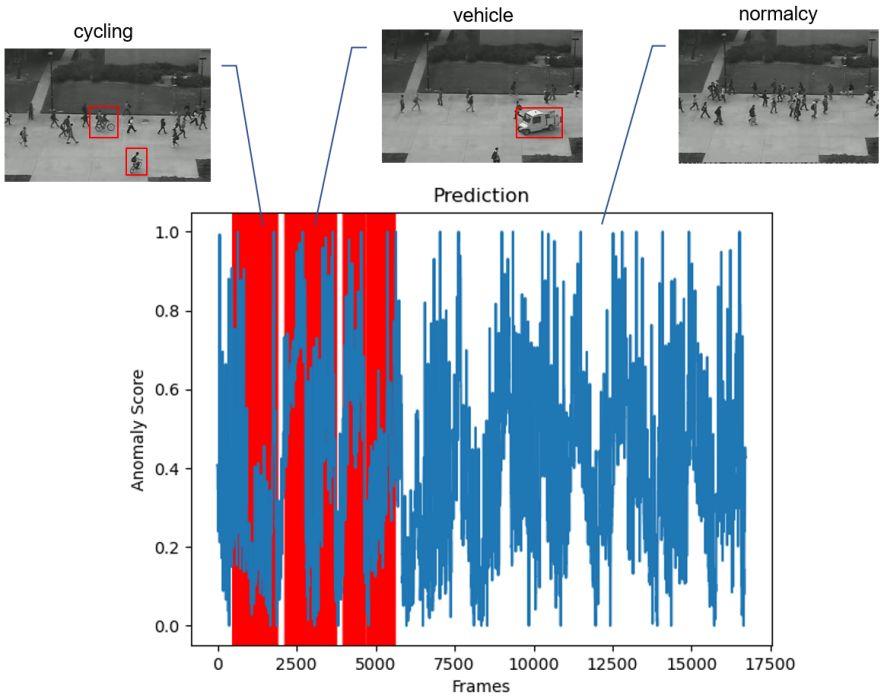


Figure 5.11: Anomaly scores produced by the ASTNet model on UCSD Ped2 Testing Set built for this dissertation.

Figure 5.12 provides a fair comparison between I3D and novel backbones such as CLIP and DinoV2 applied on Ped2 Testing Set.

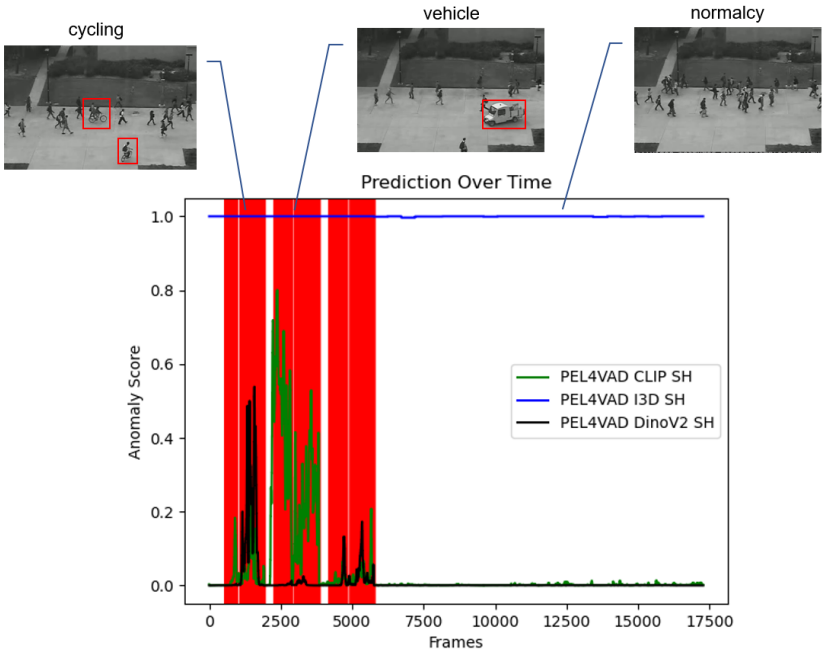


Figure 5.12: Anomaly scores produced by the PEL4VAD model on UCSD Ped2 Testing Set built for this dissertation. The model is applied using the I3D backbone and the other 2 backbones with better results (CLIP and DinoV2).

PEL4VAD CLIP SH, the best performing model, obtained 2 thresholds, and the results are presented in table 5.8.

Model	**Method**	**Prec.** (%)	**Rec.** (%)	**Acc.** (%)	**F1** (%)	**FAR** (%)
PEL4VAD CLIP SH	1*	75.52	**92.00**	90.81	**82.95**	9.57
	2*	**91.93**	71.62	**91.57**	80.51	**2.02**

Table 5.8: Comparison of different threshold values for the PEL4VAD model with CLIP backbone on UCSD Ped2 Testing Set.

In Figures 5.13 and 5.14, the green shaded area reveals the anomaly detection area when applying the 2 suggested thresholds: 0.42 % and 1.40 %.

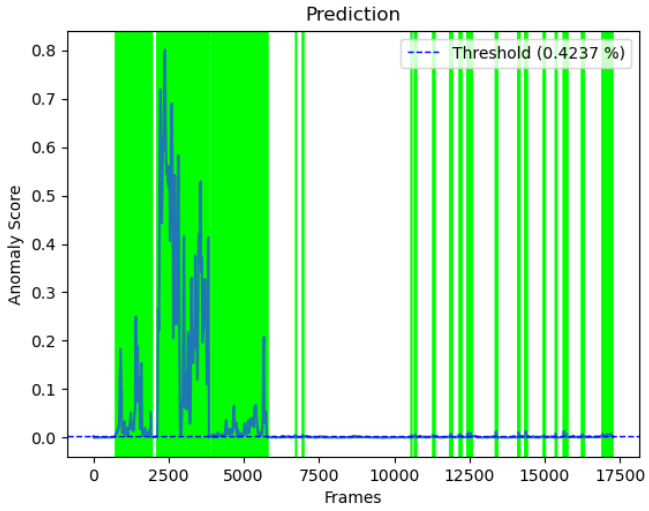


Figure 5.13: Anomaly detection produced by the PEL4VAD CLIP SH model on UCSD Ped2 Testing Set using method 1.

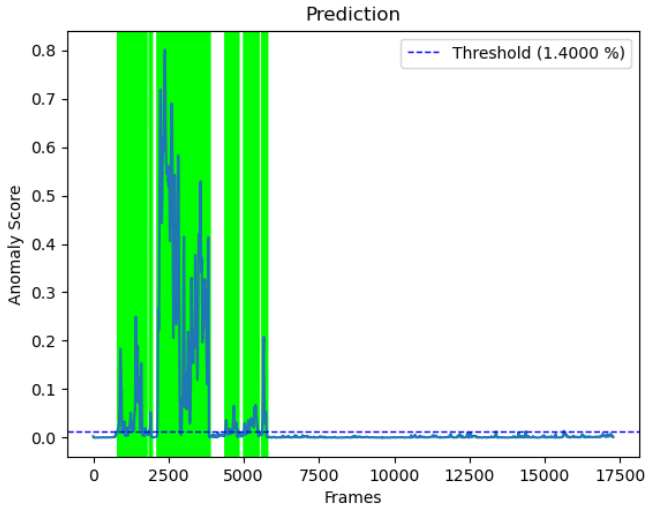


Figure 5.14: Anomaly detection produced by the PEL4VAD CLIP SH model on UCSD Ped2 Testing Set using method 2.

5.2.3 Evaluation on Adoc

The models trained on UCF and SH were tested on Adoc Testing Set with the following results in tables 5.9 and 5.10, respectively.

The Figures 5.15, 5.16 and 5.17 show the anomaly scores produced throughout Adoc Testing Set by CLIP, NextViT and I3D backbones, respectively, in order to observe their magnitude differences.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT UCF	**85.57**	**52.75**	18.14	0.00
	CLIP UCF	83.72	44.83	37.53	0.48
	DinoV2 UCF	81.46	27.05	**40.04**	**0.00**
	I3D UCF	17.65	5.91	18.14	76.57
ASTNet (SS)	WiderResnet UCF	46.78	14.28	28.80	67.56

Table 5.9: Comparison of the results achieved on Adoc dataset for the benchmark of the models trained on UCF-Crime using ASTNet and PEL4VAD models with different backbones.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT SH	**97.36**	63.03	18.14	**0.00**
	CLIP SH	96.35	**72.73**	18.14	**0.00**
	DinoV2 SH	50.86	9.09	18.14	**0.00**
	I3D SH*	95.03	62.91	**76.77**	4.49
ASTNet (SS)	WiderResnet SH	54.23	13.45	24.61	50.14

Table 5.10: Comparison of the results achieved on Adoc dataset for the benchmark of the models trained on ShanghaiTech using ASTNet and PEL4VAD models with different backbones.

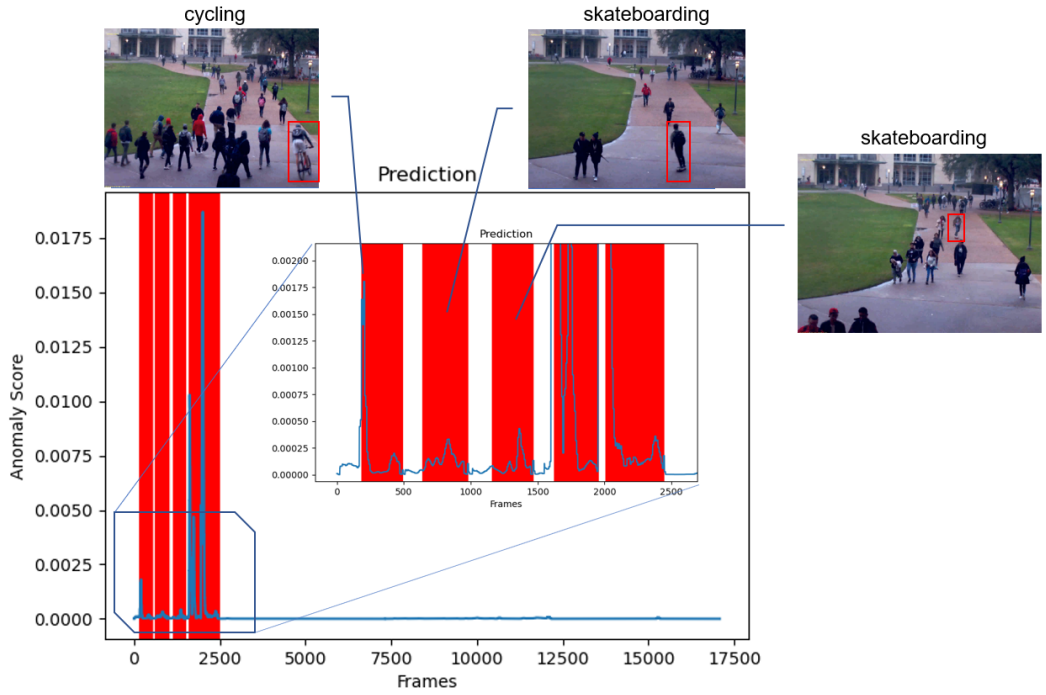


Figure 5.15: Anomaly scores produced by the PEL4VAD CLIP SH model on Adoc Testing Set.

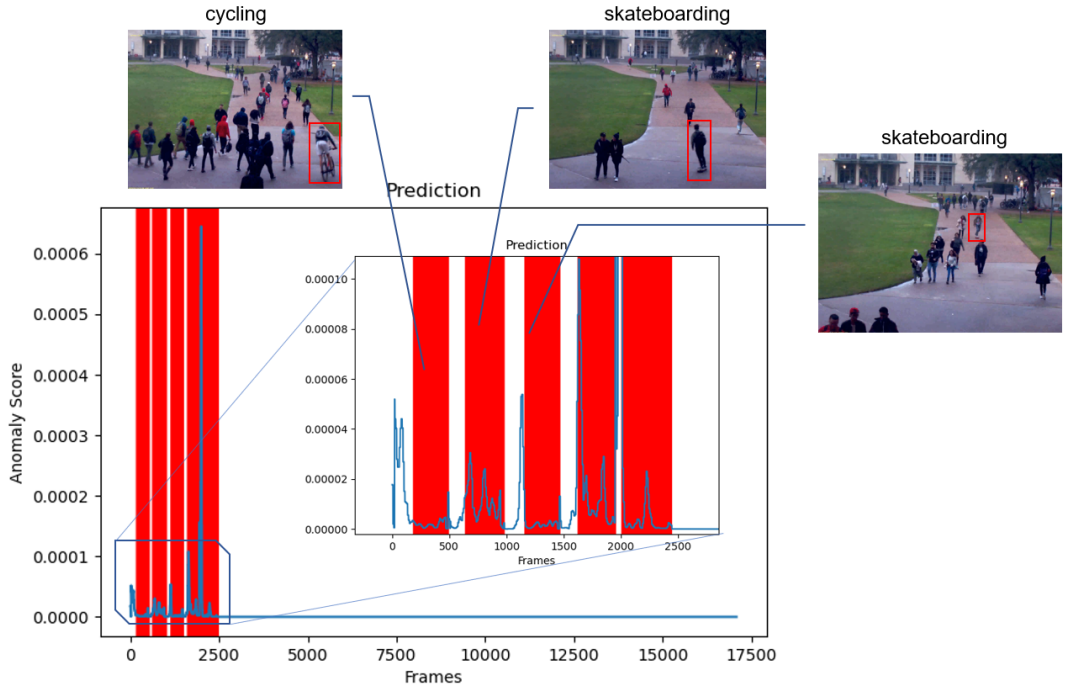


Figure 5.16: Anomaly scores produced by the PEL4VAD NextViT SH model on Adoc Testing Set.

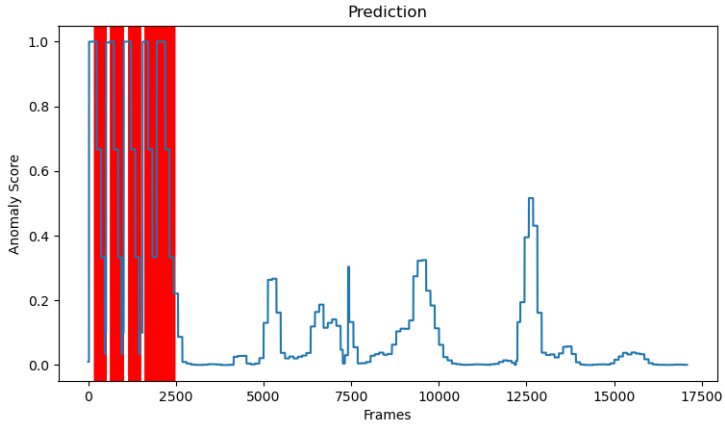


Figure 5.17: Anomaly scores produced by the PEL4VAD I3D SH model on Adoc Testing Set.

Table 5.11 compares the metrics produced by the PEL4VAD models ran by CLIP and I3D backbones at different thresholds. PEL4VAD I3D SH model seems to be the best performing model for this Testing Set.

Model	**Method**	**Prec.** (%)	**Rec.** (%)	**Acc.** (%)	**F1** (%)	**FAR** (%)
PEL4VAD I3D SH	1*	63.75	**96.48**	**94.17**	**76.77**	6.08
	2	65.78	64.20	93.10	64.98	3.70
PEL4VAD CLIP SH	1	51.37	91.31	90.51	65.75	9.58
	2	**83.33**	28.17	92.28	42.11	**0.62**

Table 5.11: Comparison of different threshold values for the PEL4VAD model with 2 backbones: I3D and CLIP.

In the graphics of the Figures 5.18, 5.19, 5.20 and 5.21, the green shaded areas show the anomaly detection regions classified by the PEL4VAD model with I3D and CLIP backbones on Adoc Testing Set, with the proposed thresholds. The 5.18 and 5.20 plots tend to miss classify some instances of normalcy while 5.19 and 5.21 have a higher precision.

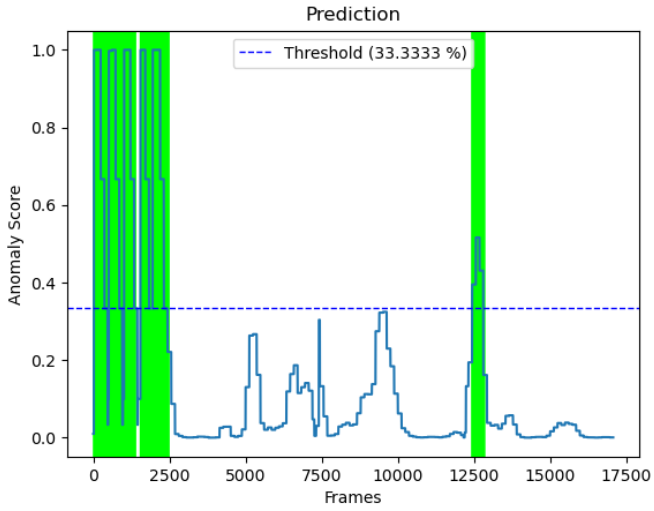


Figure 5.18: Anomaly detection produced by the PEL4VAD I3D SH model on Adoc Testing Set using method 1.

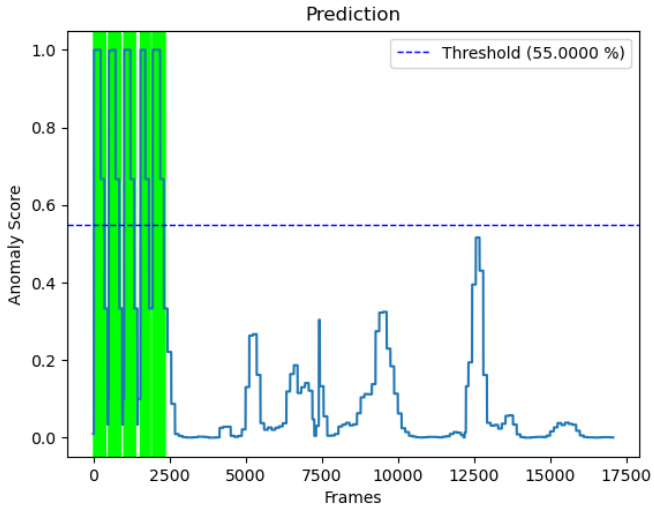


Figure 5.19: Anomaly detection produced by the PEL4VAD I3D SH model on Adoc Testing Set using method 2.

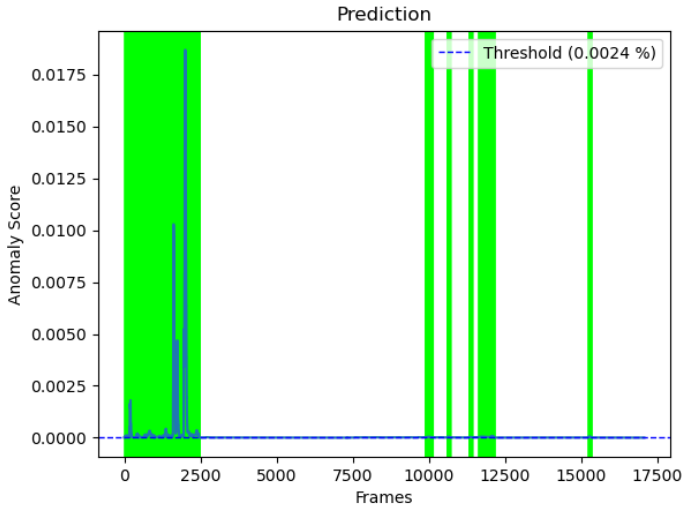


Figure 5.20: Anomaly detection produced by the PEL4VAD CLIP SH model on Adoc Testing Set using method 1.

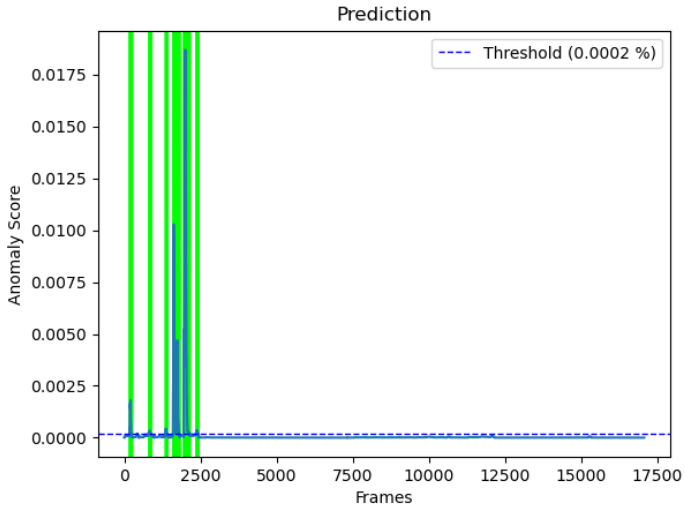


Figure 5.21: Anomaly detection produced by the PEL4VAD CLIP SH model on Adoc Testing Set using method 2.

5.2.4 Evaluation on IITB-Corridor

In tables 5.12 and 5.13, it is shown the metrics produced by the models when trained on UCF and SH, respectively.

The graphics shown in Figure 5.22 and 5.23 shows the frame-by-frame anomaly scores of the PEL4VAD and ASTNet models, where substantial differences can be seen.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT UCF	34.73	24.88	49.51	**0.00**
	CLIP UCF	**76.10**	70.82	49.51	**0.00**
	DinoV2 UCF	66.79	51.64	49.51	**0.00**
	I3D UCF	57.87	48.67	**52.19**	56.44
ASTNet (SS)	WiderResnet UCF	73.32	**80.70**	49.90	0.24

Table 5.12: Comparison of the results achieved on IITB-Corridor dataset for the benchmark of the models trained on UCF-Crime using ASTNet and PEL4VAD models with different backbones.

Model	Backbone	AUC		F1-score (%)	FAR (%)
		ROC (%)	P-R (%)		
PEL4VAD (WS)	NextViT SH	44.50	31.91	49.51	**0.00**
	CLIP SH	**83.81**	72.35	**56.61**	**0.00**
	DinoV2 SH	44.71	30.43	49.51	**0.00**
	I3D SH	49.51	30.02	53.71	52.57
ASTNet (SS)	WiderResnet SH	65.08	**73.59**	49.90	3.32

Table 5.13: Comparison of the results achieved on IITB-Corridor dataset for the benchmark of the models trained on ShanghaiTech using ASTNet and PEL4VAD models with different backbones.

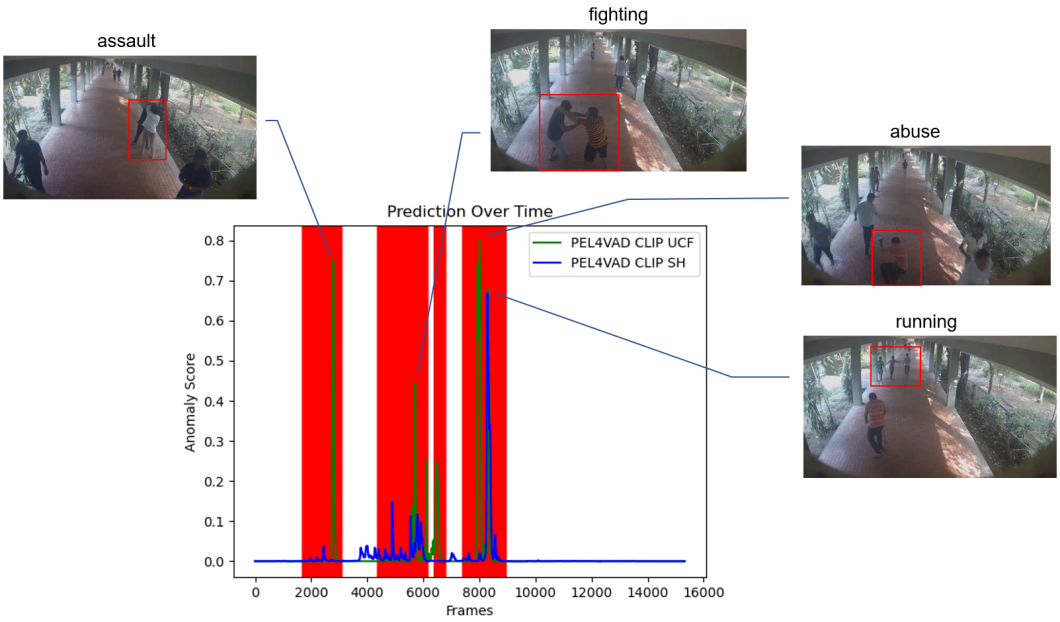


Figure 5.22: Anomaly scores produced by the PEL4VAD CLIP models on IITB-Corridor Testing Set when trained on UCF and Shanghai datasets.

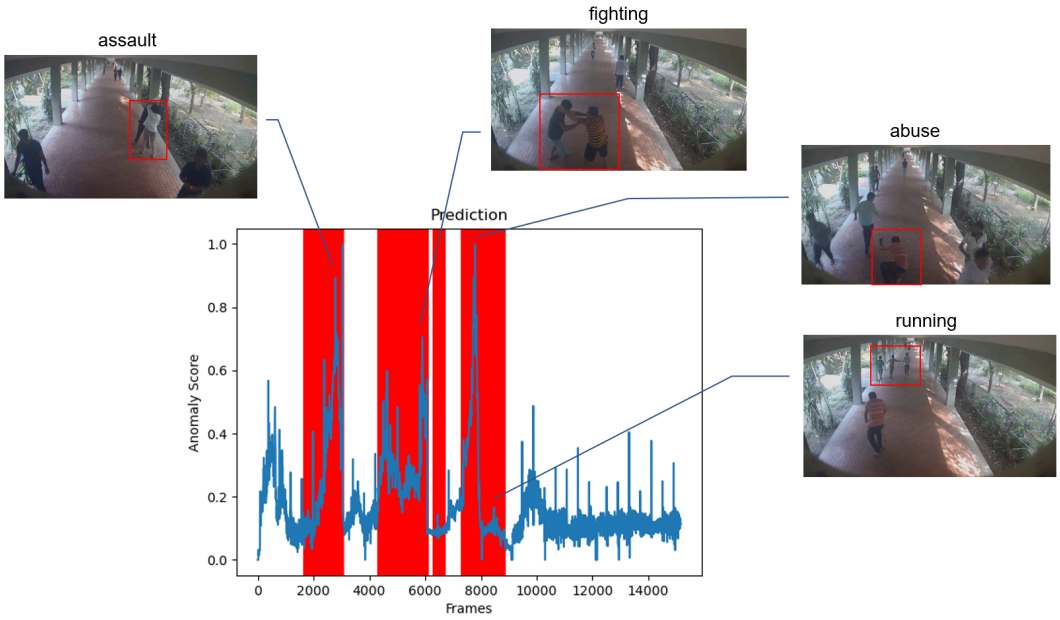


Figure 5.23: Anomaly scores produced by the ASTNet UCF model on IITB-Corridor Testing Set.

The table 5.14 presents the results for the thresholds obtained in each of the best performing models following the 2 proposed methods.

Model	**Method**	**Prec.** (%)	**Rec.** (%)	**Acc.** (%)	**F1** (%)	**FAR** (%)
PEL4VAD CLIP UCF	1	61.25	66.35	75.12	63.70	20.58
	2*	89.86	34.29	77.11	49.63	1.90
PEL4VAD CLIP SH	1*	61.55	**79.80**	76.96	**69.50**	24.44
	2	93.10	12.86	71.02	22.59	**0.47**
ASTNet UCF	1*	70.67	61.59	**78.73**	65.82	12.73
	2	**94.37**	21.27	73.40	34.72	0.63

Table 5.14: Comparison of different threshold values for the PEL4VAD models trained on UCF and Shanghai, and ASTNet model trained on UCF.

The Figures 5.24, 5.25 and 5.26 show the results when applying the suggested thresholds to determine the anomaly instances, the green shaded areas show the anomaly detection regions classified by the models.

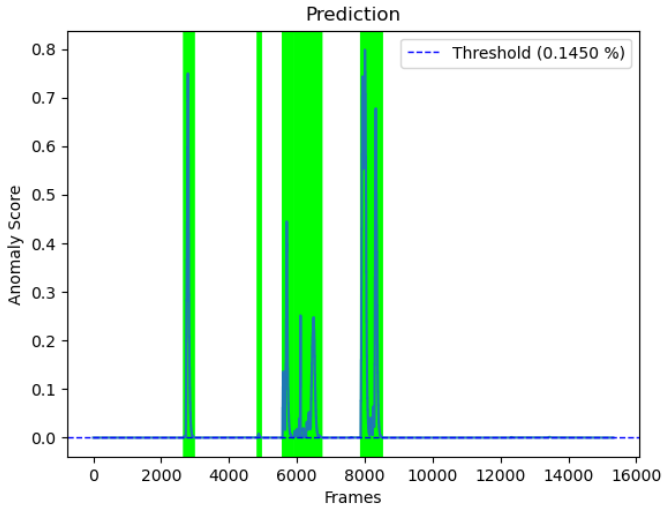


Figure 5.24: Anomaly detection produced by the PEL4VAD CLIP UCF model on IITB-Corridor Testing Set using method 2.

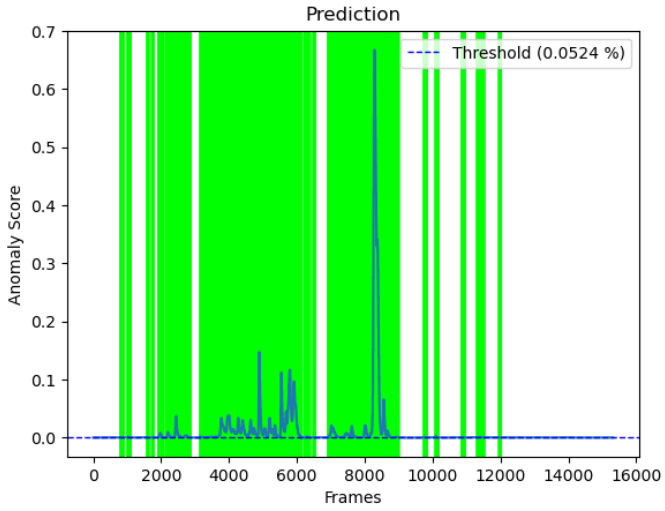


Figure 5.25: Anomaly detection produced by the PEL4VAD CLIP SH model on IITB-Corridor Testing Set using method 1.

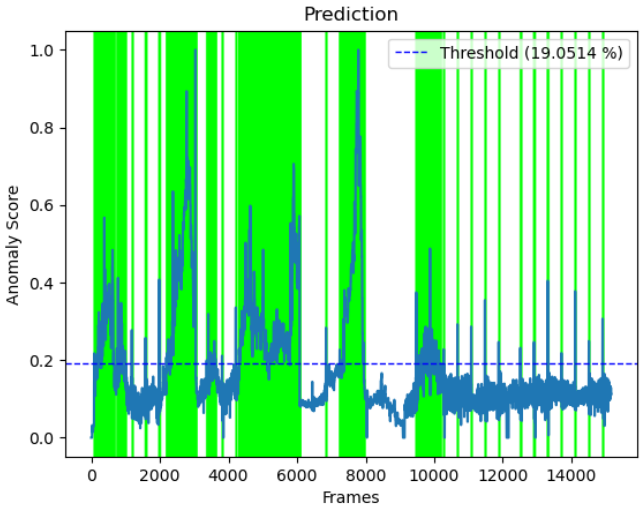


Figure 5.26: Anomaly detection produced by the ASTNet UCF model on IITB-Corridor Testing Set using method 1.

Chapter 6

Discussion

This chapter discusses the results obtained in chapter 5. The employed backbones are analysed and the different approaches (SS and WS) are also compared throughout the testing sets.

6.1 Intra-Dataset Evaluation

During this evaluation, the models operate in similar training and testing conditions, in this case using Shanghai and UCF-Crime datasets.

PEL4VAD I3D achieved the most stable anomaly scores (highest ROC) of all models regardless of the dataset applied. The ROC curves of Figure 5.1 indicate that other backbones - CLIP, DinoV2 and NextViT - generate more false positives to obtain a better detection accuracy, while I3D can detect true anomalies keeping low false positives.

CLIP backbone provides fewer false alarms than I3D when applied in more challenging datasets (in this case, UCF) as follows in Table 5.2. The SS approach via ASTNet, on the other hand, is not viable for complex features: a significantly higher variety of scenes makes harder for the SS model to learn a pattern in normal scenarios from the training.

When training on simpler scenarios (in this case, SH), the SS approach shows decent ability on detecting actual anomalies, reaching 77.31 % of P-R (Table 5.1), but it is not possible to select a threshold that takes advantage of this high score, with a F1-Score of 60.00 %.

6.2 Cross-Dataset Evaluation

In this evaluation, the backbones demonstrate considerable differences in their ability to generalize beyond the training data. The results show how each model manages OOD data.

I3D struggles with anomaly detection in crowded environments, showing higher false alarm rates than other backbones, while CLIP, DinoV2, and NextViT perform more robustly (Tables 5.4 and 5.7).

The novel backbones react differently to the scenarios in this study. CLIP consistently achieves great ROC and P-R scores regardless of the similarities between training and testing conditions. It also is capable of performing novelty detection in some cases, identifying anomalies not present in the training domain. In Table 5.22, CLIP is able to detect the fighting anomaly when trained on Shanghai, even though it is not contained in this training set.

On the other hand, DinoV2 shows high performance volatility when applied in the testing sets, with or without similar anomalies. Environmental and external conditions seem to highly affect this backbone.

NextViT do not seem to capture or learn from features directly from anomalous activities, as seen on UCSD Ped1 Testing Set, where this backbone delivers strong results when trained on the UCF Training Set, which does not contain the anomalies found in the testing set (Table 5.3). It is also possible to notice that NextViT and I3D produce similar results when trained on simpler features (in this case, from SH dataset) but when trained on more complex features such as UCF, their performance are not aligned.

In fact, I3D displayed an unique performance pattern when trained on more complex features (in this case, from UCF dataset), improving its performance when other backbones dropped and vice-versa.

The novel backbones generate much lower anomaly score magnitudes compared to I3D when applied to certain testing sets, as Figure 6.1 suggests. Camera positioning and scene features could affect those values.

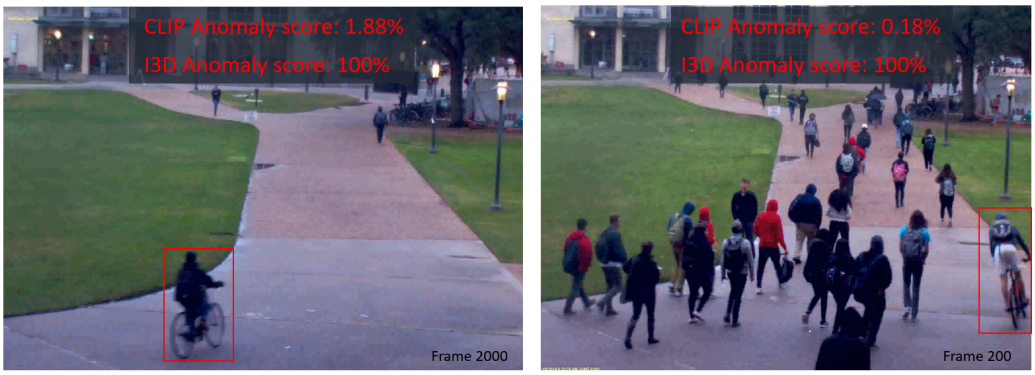


Figure 6.1: Anomaly scores produced by PEL4VAD model using I3D and CLIP backbones in 2 different anomaly frames: a) Clean Scenario where the bicycle features are seen in profile; b) Scenario where a large crowd occupies the sidewalk and the bicycle is seen from behind.

ASTNet produces noisy anomaly scores when applied on challenging scenarios, producing low ROC values in all scenarios except IITB. In contrast with other testing sets, the IITB Testing Set provides a less crowded and partial-outdoor scenario with fewer variations. Under this conditions, the ASTNet SS approach is better than PEL4VAD WS approach at detecting anomalies, even competing with the CLIP backbone (tables 5.12 and 5.13).

ASTNet improved its metrics on IITB Testing Set when fed with more and less complex features (in this case, SH and UCF features), showing adaptability under certain scenarios.

When choosing the optimal threshold to detect the anomalies, the novel backbones - CLIP, DinoV2 and NextViT - take more advantage of method 2 than I3D by achieving a much higher precision and less false alarms, as table 5.11 suggests.

6.3 Analysis of Backbones and Approaches

This section presents the performance of each backbone applied on PEL4VAD through ROC, P-R and FAR metrics across the different testing sets. It is also displayed the performance variation of PEL4VAD I3D model and ASTNet, in order to compare the different approaches.

6.3.1 Battle of Backbones

Figures 6.2 and 6.3 show the ROC and P-R scores produced by PEL4VAD model when trained on UCF and SH features, respectively.

When extracting features with CLIP, the model adapts to various scenarios without compromising its performance in anomaly detection. Specially on complex UCF features, Figure A.2 shows more stabilized anomaly scores and greater detection

accuracy when compared with I3D (higher ROC and P-R). DinoV2 shows some improvement over I3D with UCF features but its volatily is undesirable for the task of anomaly detection.

NextViT obtains barely equal ROC and P-R scores on SH features when compared with I3D, the only significant improvement was on UCSD Ped1 Testing Set with a plus 16.03 % of P-R score in A.1. When leveraging complex UCF features, NextViT surpasses I3D in half of testing sets and is outperfomred by the other half in Figure A.2, showing no overall improvement.

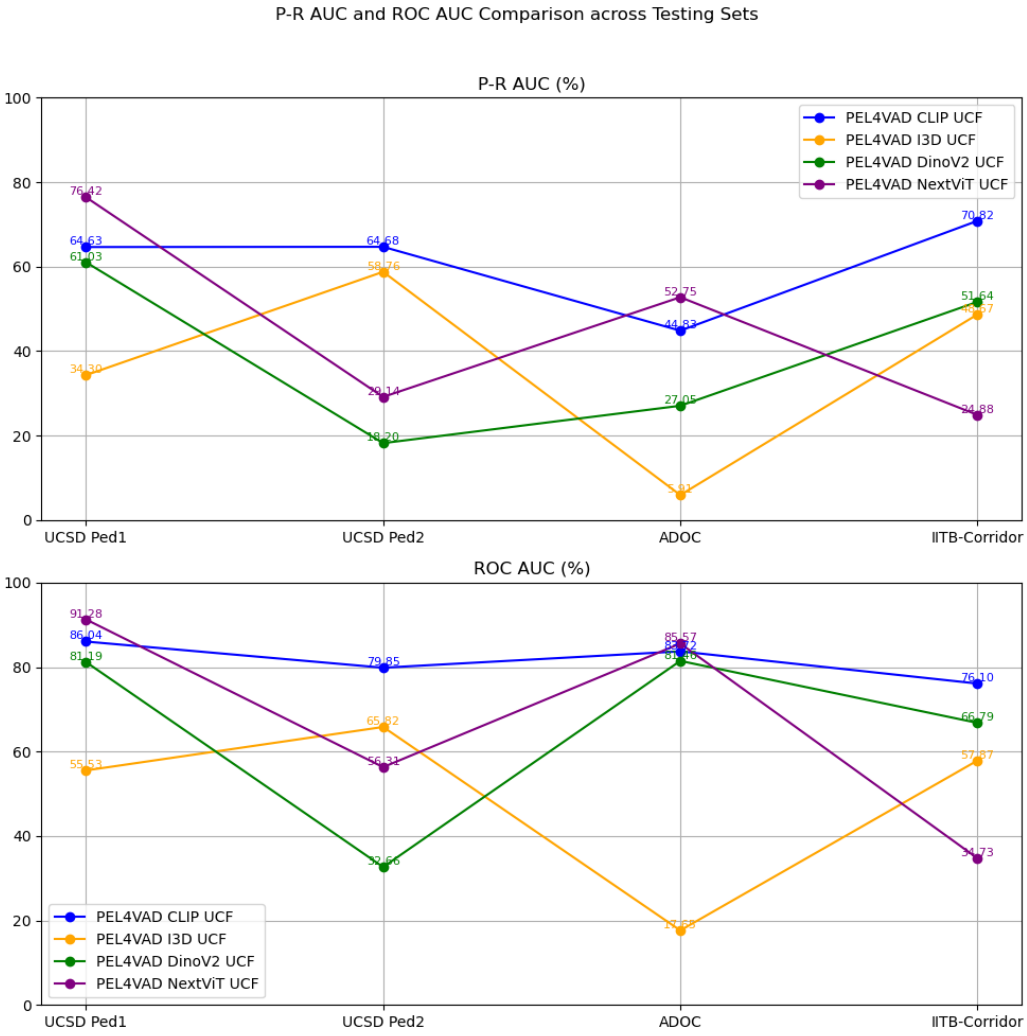


Figure 6.2: PEL4VAD performance measured by the area under the ROC and Precision-Recall curves across different backbones when trained on UCF Training Set.

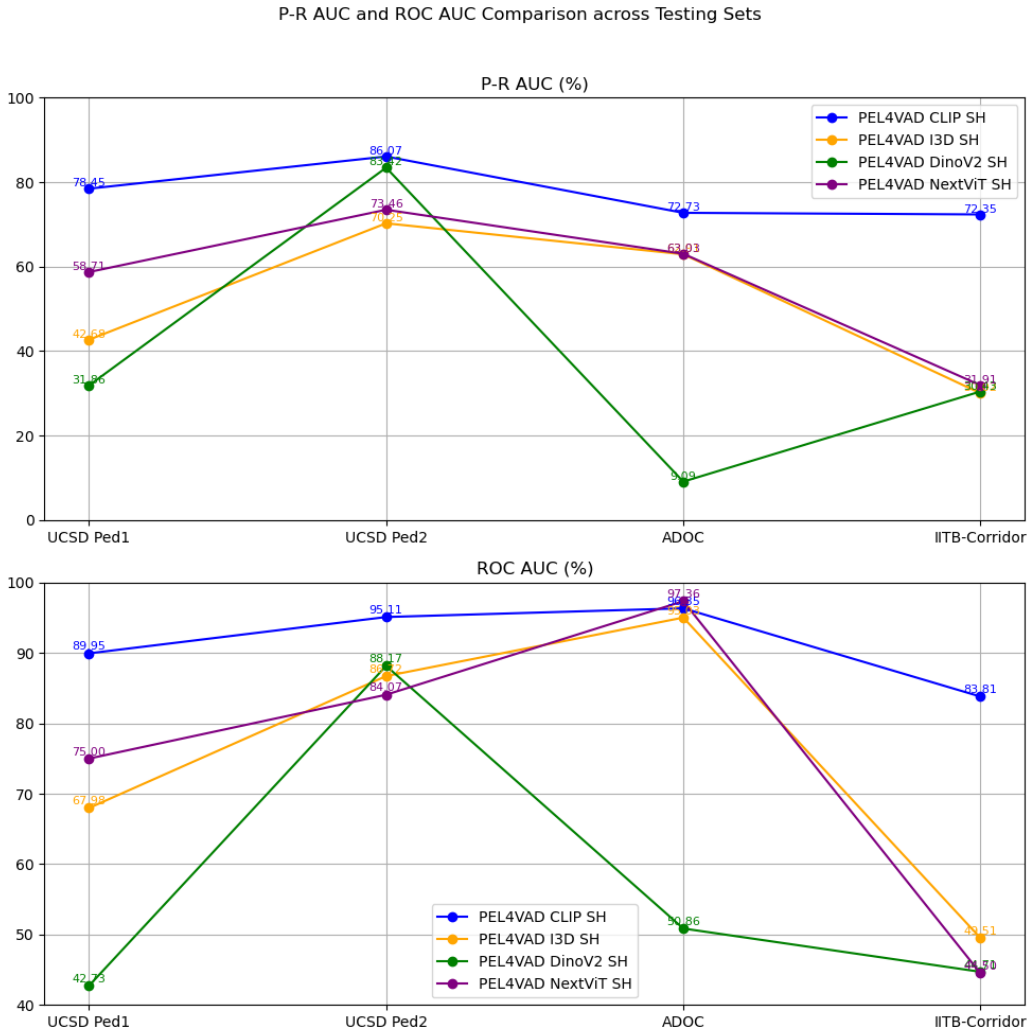


Figure 6.3: PEL4VAD performance measured by the area under the ROC and Precision-Recall curves across different backbones when trained on SH Training Set.

The Figures 6.4 and 6.5 show the FAR produced by PEL4VAD model when trained on UCF and SH, respectively.

The FAR produced by novel backbones are considerably lower than I3D when trained on UCF and SH features.

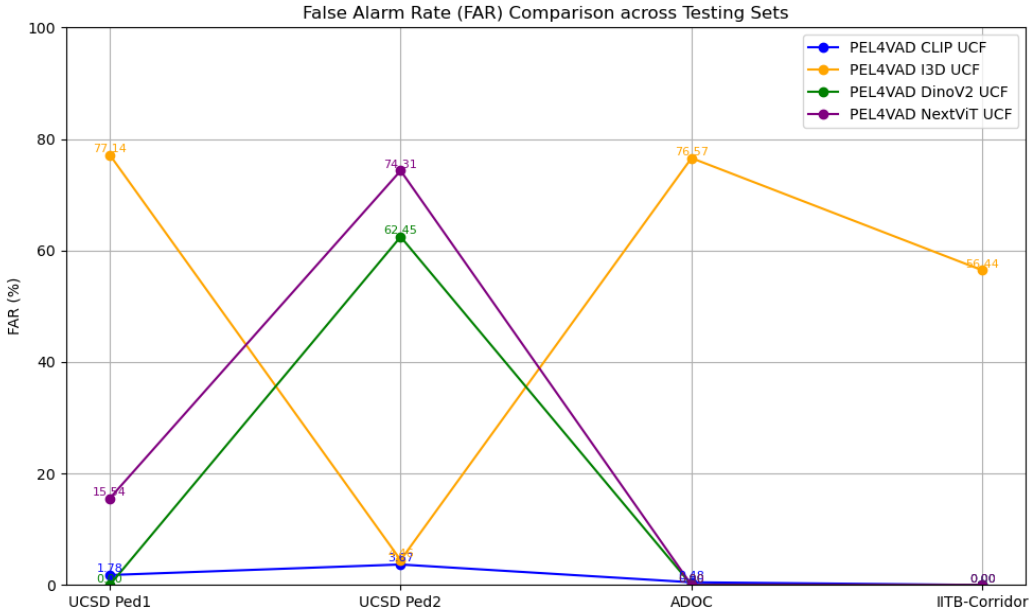


Figure 6.4: PEL4VAD performance measured by FAR metric across different backbones when trained on UCF Training Set.

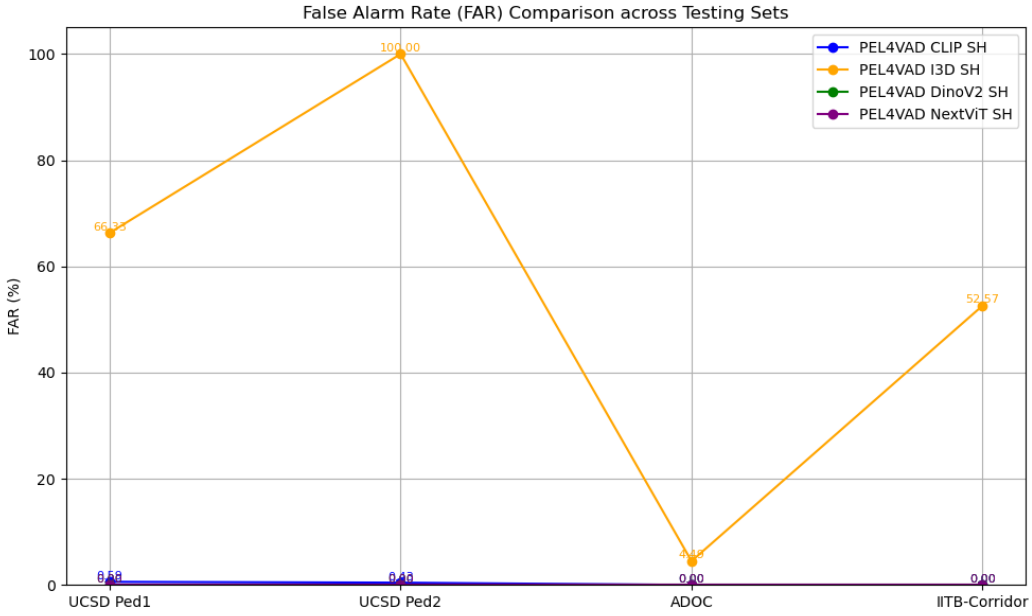


Figure 6.5: PEL4VAD performance measured by FAR metric across different backbones when trained on SH Training Set.

6.3.2 Battle of Supervision Approaches

The Figures 6.6 and 6.7 show the ROC and P-R scores produced by both models PEL4VAD I3D and ASTNet when trained using UCF and SH features, respectively.

OOC solutions like ASTNet are very sensitive to environmental changes, this model performed badly even when there are similar anomalies between training and testing sets in Figure A.3.

The effort needed to train a model for anomaly detection depends on the type of approach used. With SS the model leverages normal videos for training so it is only necessary to record a similar environment without the occurrence of an anomaly, which is effortless and does not require any labelling. The training process in WS require anomalies to be identified, and is more robust in challenging scenarios. Therefore, SS approach obtains greater generalization capacity with much less effort, obtaining more 9.88 % of P-R score than WS approach and a similar ROC score when PEL4VAD leverages the robust CLIP backbone in Figure A.5.

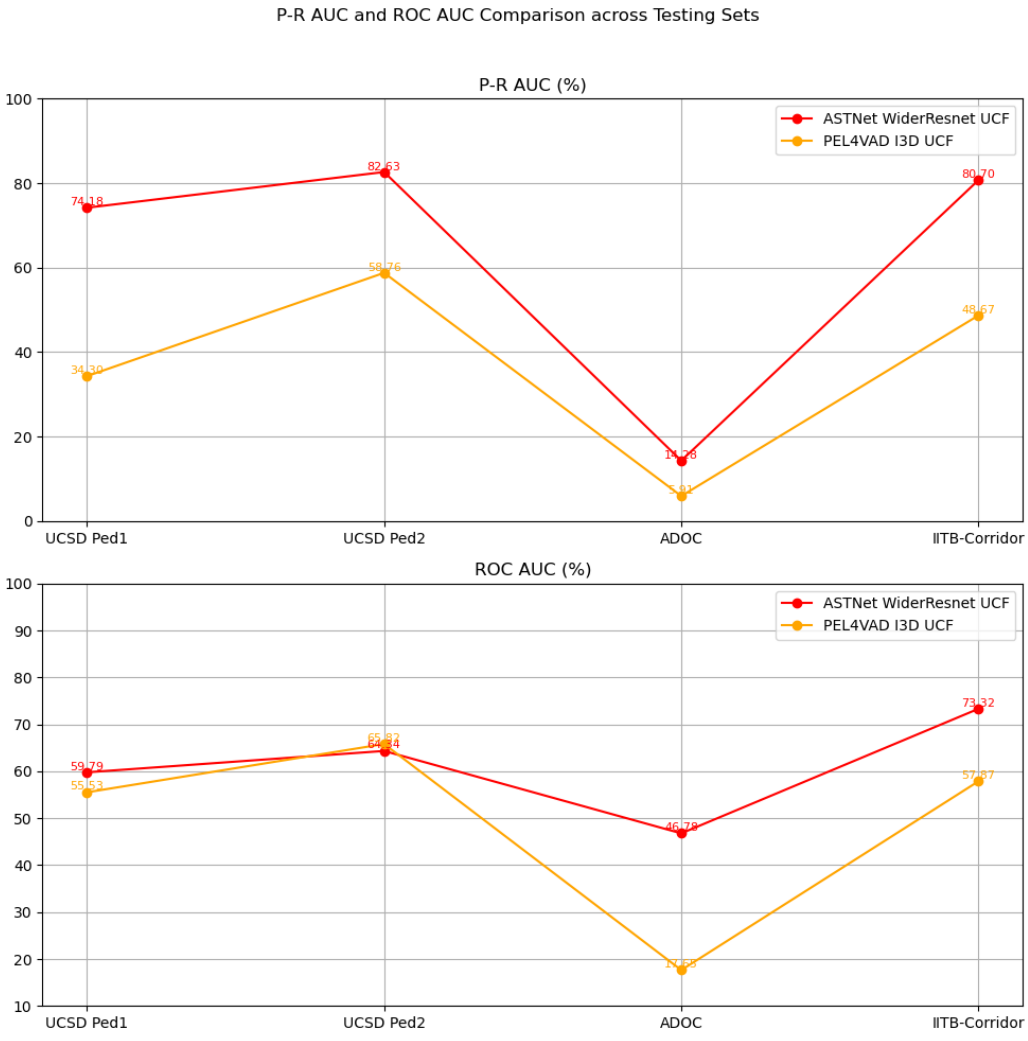


Figure 6.6: PEL4VAD and ASTNet performances measured by the area under the ROC and Precision-Recall curves when trained on UCF Training Set.

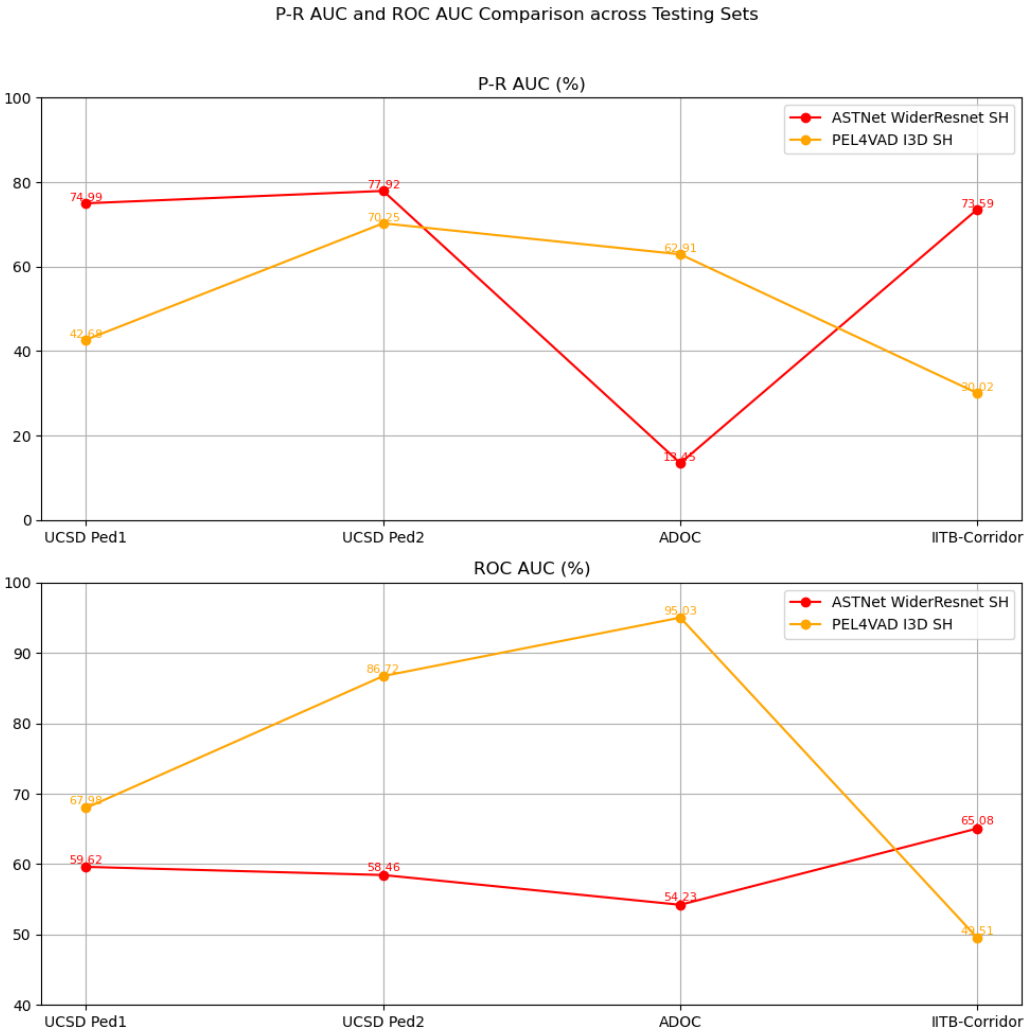


Figure 6.7: PEL4VAD and ASTNet performances measured by the area under the ROC and Precision-Recall curves when trained on SH Training Set.

The Figures 6.8 and 6.9 show the FAR produced by each approach through ASTNet and PEL4VAD I3D models.

WS approach generates more random and higher false alarms when comparing to SS, which shows the same patterns regardless of the training features.

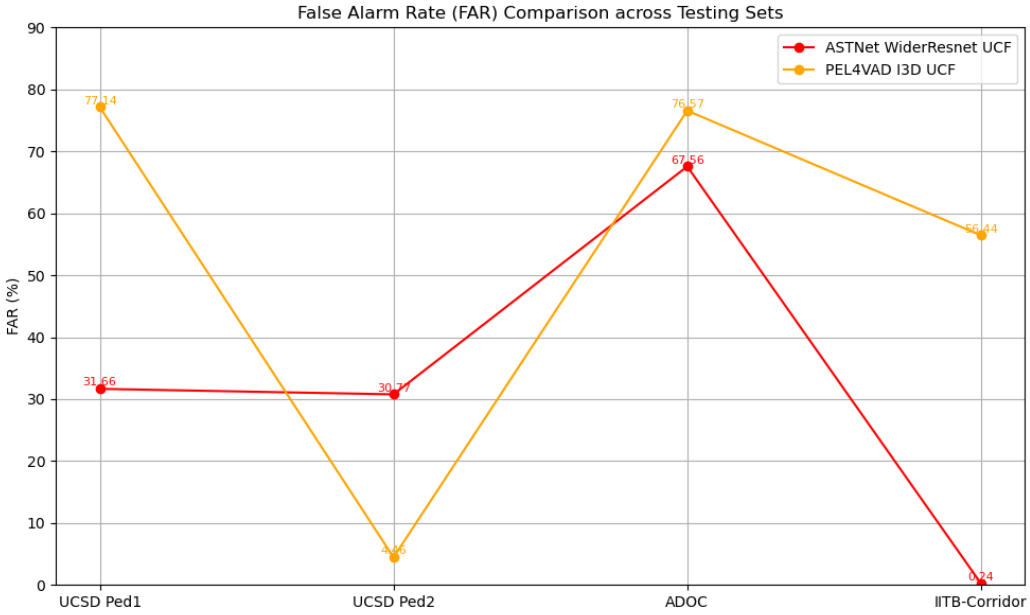


Figure 6.8: PEL4VAD and ASTNet performances measured by FAR metric when trained on UCF Training Set.

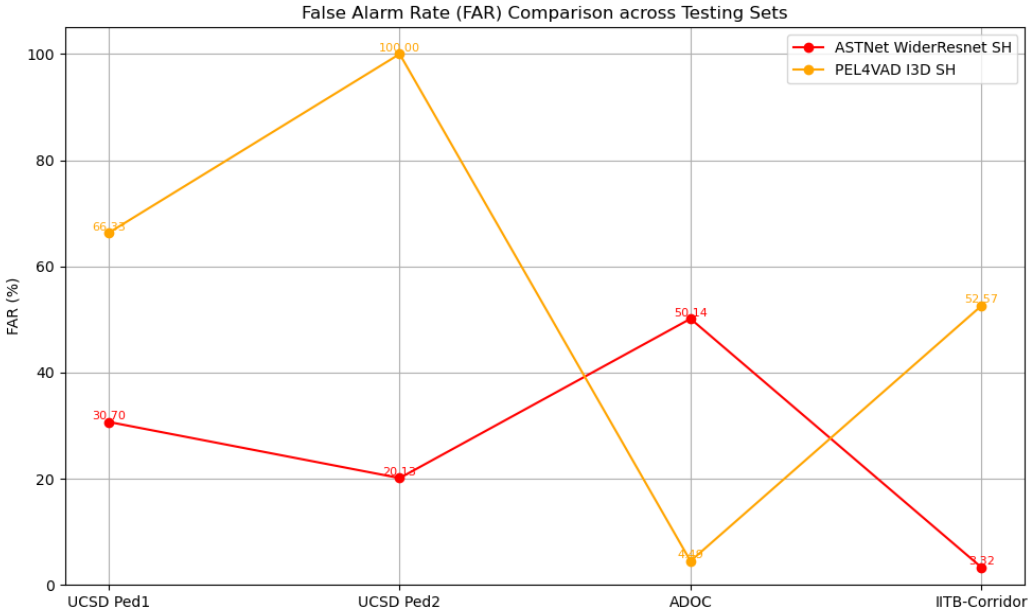


Figure 6.9: PEL4VAD and ASTNet performances measured by FAR metric when trained on SH Training Set.

Chapter 7

Conclusion

This chapter summarizes the key-points taken from the experiments. It covers the two main aspects of this work: analyse the impact of emerging backbones on anomaly detection and understand the generalization capacity of SS and WS approaches.

7.1 New Findings

In this work, the selected backbones represent some of the most recent and innovative architecture types and supervised learning techniques, traditionally applied in computer vision tasks, but now adapted for VAD. Specifically, the CLIP, DinoV2, and NextViT backbones are implemented to explore their performance in this domain.

Notably, the integration of these emerging backbones was done in a raw manner, with no modifications or adaptations to the existing VAD model architecture. The results are purely obtained from feeding the model with features extracted by the backbones, allowing a direct comparison of their effectiveness. Based on the characteristics of each backbone, several conclusions can be drawn.

CLIP demonstrated the strongest generalization capabilities across different environments, outperforming the other backbones in handling unseen data. In contrast, DinoV2 was highly sensitive to variations in the scenario conditions. A comparison between these 2 SSL backbones highlights the potential for future advancements in anomaly detection through multi-modal SSL techniques.

This study also concludes that the new supervised backbone architectures, such as NextViT, are still unable to fully address the limitations of traditional supervised backbones like I3D.

When testing models on the same dataset they were trained on, I3D achieved the best results, with the fewest in-distribution errors. However, when faced with OOD challenges, I3D was surpassed by more advanced backbones, particularly CLIP.

Even though SS approach do not seem to benefit from novel backbones as WS, this study concludes that SS shows greater generalization capacity on more restrictive scenarios and there's advantages on choosing SS over WS on simpler environments.

7.2 Future Work

The implementation of these new backbones have made substantial improvements in generalization without requiring any changes in the architecture of anomaly detection models.

However, when feeding the model with features extracted from novel backbones, it was found the presence of anomaly score normalization issues. These issues likely stem from the lack of architectural adaptations in the VAD model to fully integrate the novel features extracted by the emerging backbones. Therefore, in-depth changes in the architecture of models may help to overcome these obstacles in the future.

In the next step, it is intended to implement Few-shot Learning, which aims to adapt quickly to a new task with only a few training samples. This works by training a model using video data collected from various camera videos within the same scenario. This reduces substantially the training effort and could potentially achieve better results. Another advantage is that the trained model can be adapted to novel viewpoints but its adaptability is still confined to a specific scenario.

Combining new model training techniques like Few-Shot Learning with emerging backbones explored in this dissertation could further increase the generalization capacity of these models. Current training techniques leverage different training and testing conditions, which impacts the performance of the models. Few-Shot Learning looks promising of being applied in the future.

References

[1] H. Lv and Q. Sun, "Video anomaly detection and explanation via large language models," *arXiv preprint arXiv:2401.05702*, 2024. Singapore Management University. [Cited on page 1]

[2] D. L. S. G. Wen Liu, Weixin Luo, "Future frame prediction for anomaly detection – a new baseline," tech. rep., IEEE/CVF Conf. on Computer Vision and Pattern Recognition, Shanghai, 2018. [Cited on page 2]

[3] M. Z. Zaheer, A. Mahmood, M. H. Khan, M. Segu, F. Yu, and S.-I. Lee, "Generative cooperative learning for unsupervised video anomaly detection," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14724–14734, 2022. [Cited on pages ix, 2, and 26]

[4] F. Caetano, P. Carvalho, and J. S. Cardoso, "Unveiling the performance of video anomaly detection models — a benchmark-based review," *Intelligent Systems with Applications*, vol. 18, p. 200236, 2023. [Cited on pages 2, 28, 31, 44, 52, and 53]

[5] J. Guerin, K. Delmas, R. Ferreira, and J. Guiochet, "Out-of-distribution detection is not all you need," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, pp. 14829–14837, Jun. 2023. [Cited on pages ix, 3, and 4]

[6] S. Chatterjee, "What is feature extraction? feature extraction in image processing." Available at `https://www.mygreatlearning.com/blog/feature-extraction-in-image-processing/#what-is-feature-extraction`, 2022. (Last accessed in 01/05/2024). [Cited on page 7]

[7] G. Pai, G. Baloch, and I. Rashik, "What are the steps to extract features from an image?." Available at `https://www.linkedin.com/advice/1/what-steps-extract-features-from-image-dttmc?lang=en`, 2023. (Last accessed in 01/05/2024). [Cited on page 8]

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017. [Cited on page 8]

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021. [Cited on pages ix and 9]

[10] D. Shah, "Vision transformer: What it is how it works [2023 guide]." Available at `https://www.v7labs.com/blog/vision-transformer-guide`, 2022. (Last accessed in 04/05/2024). [Cited on page 9]

[11] S. Khan, H. Rahmani, S. Shah, and M. Bennamoun, *A Guide to Convolutional Neural Networks for Computer Vision*. No. 1 in Synthesis Lectures on Computer Vision, Morgan Claypool Publishers, 2018. [Cited on pages ix, 10, 11, and 12]

[12] A. Padda, "How do pca and autoencoders compare in dimensionality reduction techniques?." Available at `https://www.linkedin.com/advice/1/how-do-pca-autoencoders-compare-dimensionality-osatc?lang=en`, 2024. (Last accessed in 27/05/2024). [Cited on page 12]

[13] U. Michelucci, "An introduction to autoencoders," *arXiv preprint arXiv:2201.03898*, 2022. [Cited on pages ix and 12]

[14] MathWorks, "Autoencoders." Available at `https://www.mathworks.com/discovery/autoencoder.html`, 2024. (Last accessed in 27/05/2024). [Cited on page 12]

[15] A. Anwar, "Difference between alexnet, vggnet, resnet, and inception." Available at `https://towardsdatascience.com/the-w3h-of-alexnet-vggnet-resnet-and-inception-7baaaecccc96`, 2019. (Last accessed in 03/06/2024). [Cited on pages 13 and 14]

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision – ECCV 2016* (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), (Cham), pp. 630–645, Springer International Publishing, 2016. [Cited on page 13]

[17] D. V. Sang and N. D. Minh, "Fully residual convolutional neural networks for aerial image segmentation," in *Proceedings of the 9th International Symposium on Information and Communication Technology*, SoICT '18, (New York, NY, USA), p. 289–296, Association for Computing Machinery, 2018. [Cited on pages ix and 13]

[18] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016. Université Paris-Est, École des Ponts. [Cited on pages ix, 13, and 14]

[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015. [Cited on pages ix and 14]

[20] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724–4733, 2017. [Cited on pages ix, xv, and 15]

[21] T. Huang, S. Liu, T. Chen, and Z. Wang, "The counterattack of cnns in self-supervised learning: Larger kernel size might be all you need," *arXiv preprint arXiv:2312.05695*, 2023. [Cited on pages 16 and 17]

[22] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Cited on page 16]

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Cited on page 16]

[24] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. B. Girshick, "Masked autoencoders are scalable vision learners," *CoRR*, vol. abs/2111.06377, 2021. [Cited on page 16]

[25] M. Goldblum, H. Souri, R. Ni, M. Shu, V. Prabhu, G. Somepalli, P. Chattopadhyay, M. Ibrahim, A. Bardes, J. Hoffman, *et al.*, "Battle of the backbones: A large-scale comparison of pretrained models across computer vision tasks," *Advances in Neural Information Processing Systems*, vol. 36, 2024. [Cited on pages x, 16, 17, and 27]

[26] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning* (M. Meila and T. Zhang, eds.), vol. 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763, PMLR, 18–24 Jul 2021. [Cited on pages ix, 17, 18, 19, and 20]

[27] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision," 2024. [Cited on pages ix, x, xv, 20, 21, and 22]

[28] R. Wightman, "Openclip." `https://github.com/mlfoundations/open_clip`, 2024. [Cited on page 21]

[29] J. Li, X. Xia, W. Li, H. Li, X. Wang, X. Xiao, R. Wang, M. Zheng, and X. Pan, "Next-vit: Next generation vision transformer for efficient deployment in realistic industrial scenarios," *arXiv preprint arXiv:2207.05501*, 2022. [Cited on pages x, 22, 23, and 24]

[30] Y. Pu, X. Wu, and S. Wang, "Learning prompt-enhanced context features for weakly-supervised video anomaly detection," *arXiv preprint arXiv:2306.14451*, 2023. [Cited on pages x, 26, 30, 52, and 53]

[31] J. Fioresi, I. R. Dave, and M. Shah, "Ted-spad: Temporal distinctiveness for self-supervised privacy-preservation for video anomaly detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13598–13609, 2023. [Cited on page 26]

[32] H. Joo, K. T. Vo, K. Yamazaki, and N. T. H. Le, "Clip-tsa: Clip-assisted temporal self-attention for weakly-supervised video anomaly detection," *2023 IEEE International Conference on Image Processing (ICIP)*, pp. 3230–3234, 2022. [Cited on page 26]

[33] W. Chen, K. T. Ma, Z. Jian Yew, M. Hur, and D. A.-A. Khoo, "Tevad: Improved video anomaly detection with captions," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 5549–5559, 2023. [Cited on pages x, 26, and 30]

[34] M. Cho, M. Kim, S. Hwang, C. Park, K. Lee, and S. Lee, "Look around for anomalies: Weakly-supervised anomaly detection via context-motion relational learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12137–12146, June 2023. [Cited on page 26]

[35] C. Zhang, G. Li, Y. Qi, S. Wang, L. Qing, Q. Huang, and M.-H. Yang, "Exploiting completeness and uncertainty of pseudo labels for weakly supervised video anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16271–16280, 2023. [Cited on page 26]

[36] Y. Tian, G. Pang, Y. Chen, R. Singh, J. W. Verjans, and G. Carneiro, "Weakly-supervised video anomaly detection with robust temporal feature magnitude learning," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4975–4986, 2021. [Cited on page 26]

[37] J.-C. Feng, F.-T. Hong, and W.-S. Zheng, "Mist: Multiple instance self-training framework for video anomaly detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14009–14018, 2021. [Cited on page 26]

[38] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6479–6488, 2018. [Cited on pages x, 26, 28, and 29]

[39] C. Shi, C. Sun, Y. Wu, and Y. Jia, "Video anomaly detection via sequentially learning multiple pretext tasks," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10296–10306, 2023. [Cited on page 26]

[40] Z. Yang, J. Liu, Z. Wu, P. Wu, and X. Liu, "Video event restoration based on keyframes for video anomaly detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14592–14601, 2023. [Cited on page 26]

[41] V.-T. Le and Y.-G. Kim, "Attention-based residual autoencoder for video anomaly detection," *Applied Intelligence*, vol. 53, no. 3, pp. 3240–3254, 2023. [Cited on pages x, 26, 32, and 33]

[42] T. Nguyen and J. Meunier, "Anomaly detection in video sequence with appearance-motion correspondence," *CoRR*, vol. abs/1908.06351, 2019. [Cited on page 26]

[43] B. E. C. F. Ding W, Liu K, "Future frame prediction for anomaly detection – a new baseline," tech. rep., Pattern Recognit, Xidian, 2018. [Cited on page 26]

[44] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," *CoRR*, vol. abs/1604.04574, 2016. [Cited on page 26]

[45] M. Tran, "Understanding u-net." Available at `https://towardsdatascience.com/understanding-u-net-61276b10f360`, 2022. (Last accessed in 28/05/2024). [Cited on page 26]

[46] K. Vishniakov, Z. Shen, and Z. Liu, "Convnet vs transformer, supervised vs clip: Beyond imagenet accuracy," *arXiv preprint arXiv:2311.09215*, 2023. [Cited on page 28]

[47] L. Y. S. W. Yujiang Pu, Xiaoyu Wu, "Learning prompt-enhanced context features for weakly-supervised video anomaly detection." `https://github.com/yujiangpu20/PEL4VAD/tree/master`, 2024. [Cited on pages 28 and 29]

[48] K. Y. N. L. Kevin Hyekang Joo, Khoa Vo, "Clip-tsa: Clip-assisted temporal self-attention for weakly-supervised video anomaly detection." https://github.com/joos2010kj/CLIP-TSA, 2023. [Cited on pages 28 and 29]

[49] W. Chen, K. T. Ma, Z. J. Yew, M. Hur, and D. Khoo, "Tevad: Improved video anomaly detection with captions." https://github.com/coranholmes/TEVAD, 2023. [Cited on pages 28 and 29]

[50] Y. C. R. S. J. W. V. G. C. Yu Tian, Guansong Pang, "Weakly-supervised video anomaly detection with robust temporal feature magnitude learning." https://github.com/tianyu0207/RTFM/tree/main?tab=readme-ov-file, 2021. [Cited on pages 28 and 29]

[51] D. L. Wen Liu, Weixin Luo and S. Gao, "Future frame prediction for anomaly detection – a new baseline." https://github.com/StevenLiuWen/ano_pred_cvpr2018, 2018. [Cited on page 31]

[52] V.-T. Le and Y.-G. Kim, "Astnet: Attention-based residual autoencoder for video anomaly detection." https://github.com/vt-le/astnet/tree/main, 2022. [Cited on page 31]

[53] W. Liu, D. L. W. Luo, and S. Gao, "Future frame prediction for anomaly detection – a new baseline," in *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. [Cited on pages x, 32, and 39]

[54] K. Lim, "Roc (receiver operating curve) compilation.." Available at https://rpubs.com/LIMKYUSON/681052, 2020. (Last accessed in 21/09/2020). [Cited on pages x and 37]

[55] C. for Research in Computer Vision, "Real-world anomaly detection in surveillance videos." Available at https://www.crcv.ucf.edu/projects/real-world/, 2018. (Last accessed in 05/03/2024). [Cited on page 39]

[56] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1975–1981, 2010. [Cited on page 39]

[57] P. Mantini, Z. Li, and K. S. Shah, "A day on campus - an anomaly detection dataset for events in a single camera," in *Computer Vision – ACCV 2020* (H. Ishikawa, C.-L. Liu, T. Pajdla, and J. Shi, eds.), (Cham), pp. 619–635, Springer International Publishing, 2021. [Cited on page 39]

[58] R. Rodrigues, N. Bhargava, R. Velmurugan, and S. Chaudhuri, "Multi-timescale trajectory prediction for abnormal human activity detection," in *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2020. [Cited on page 39]

[59] K. Deshpande, N. S. Punn, S. K. Sonbhadra, and S. Agarwal, "Anomaly detection in surveillance videos using transformer based attention model," in *International Conference on Neural Information Processing*, pp. 199–211, Springer, 2022. [Cited on page 52]

[60] R. Kundu, "F1 score in machine learning: Intro calculation." Available at `https://www.v7labs.com/blog/f1-score-guide`, 2022. (Last accessed in 14/06/2024). [Cited on page 52]

Appendix A

Appendix

A.1 Backbones Comparison

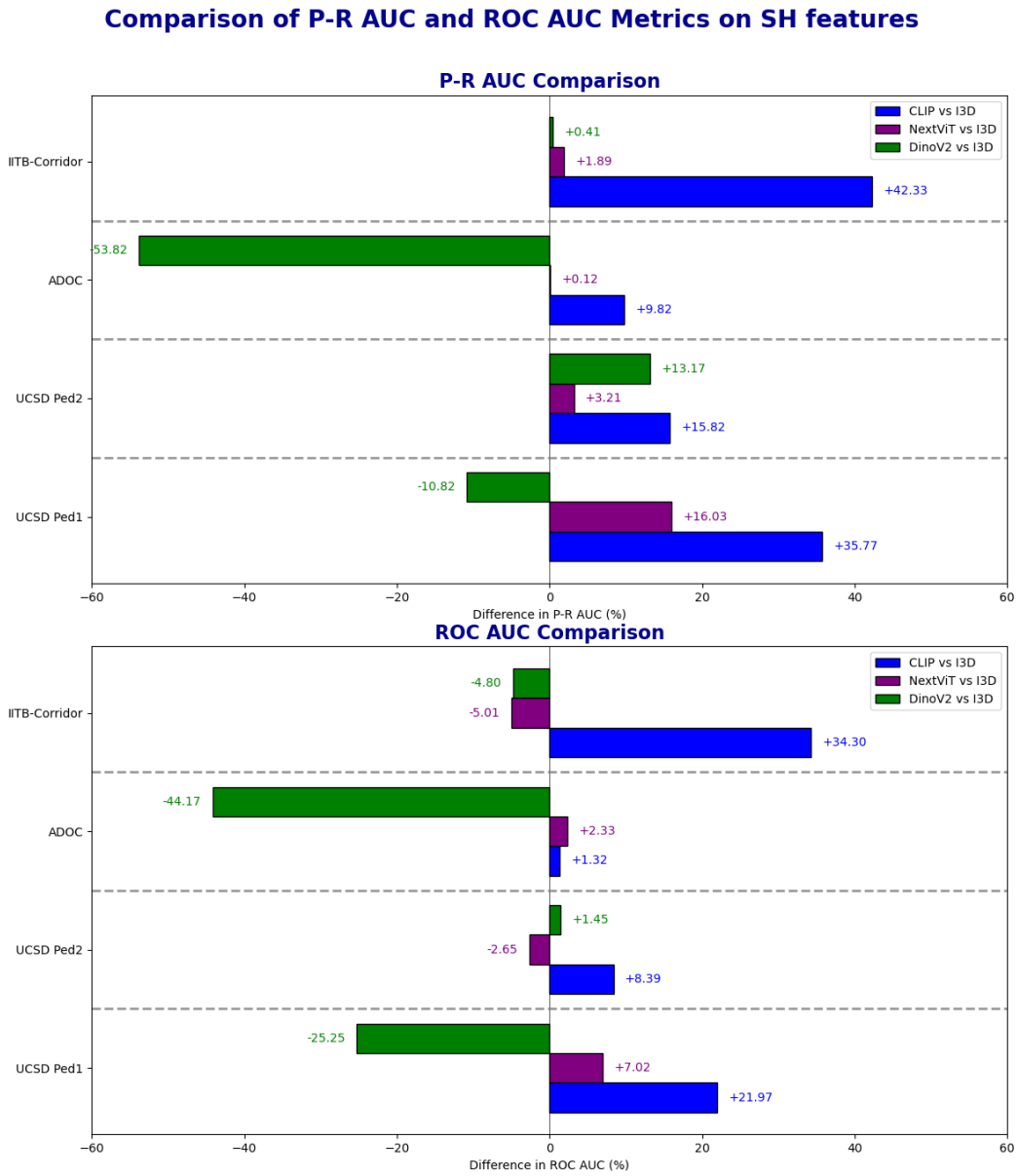


Figure A.1: P-R and ROC results compared with I3D throughout CLIP, NextViT and DinoV2 backbones. The model leveraged features extracted from SH Training Set.

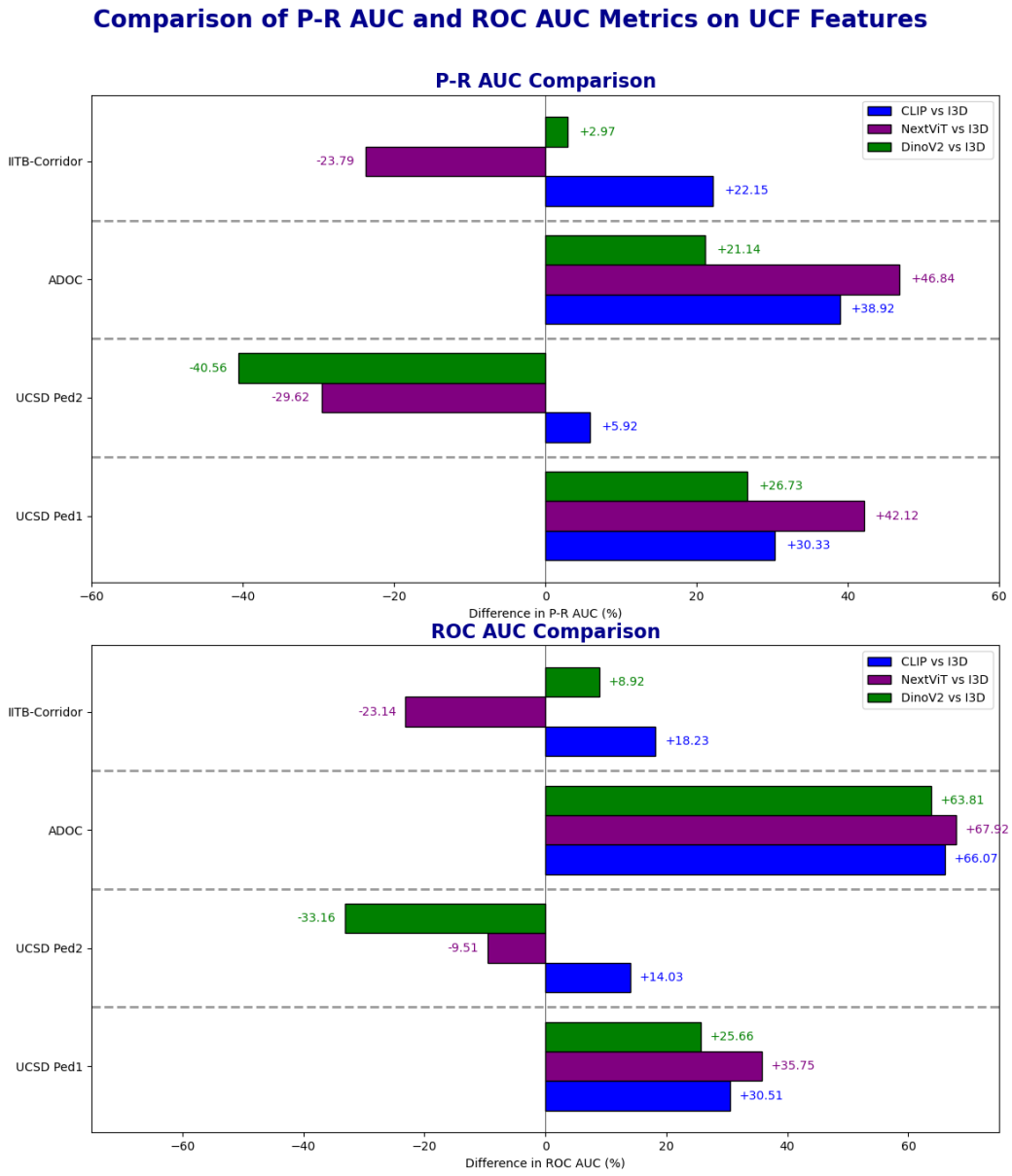


Figure A.2: P-R and ROC results compared with I3D throughout CLIP, NextViT and DinoV2 backbones. The model leveraged features extracted from UCF Training Set.

A.2 Supervision Approaches Comparison

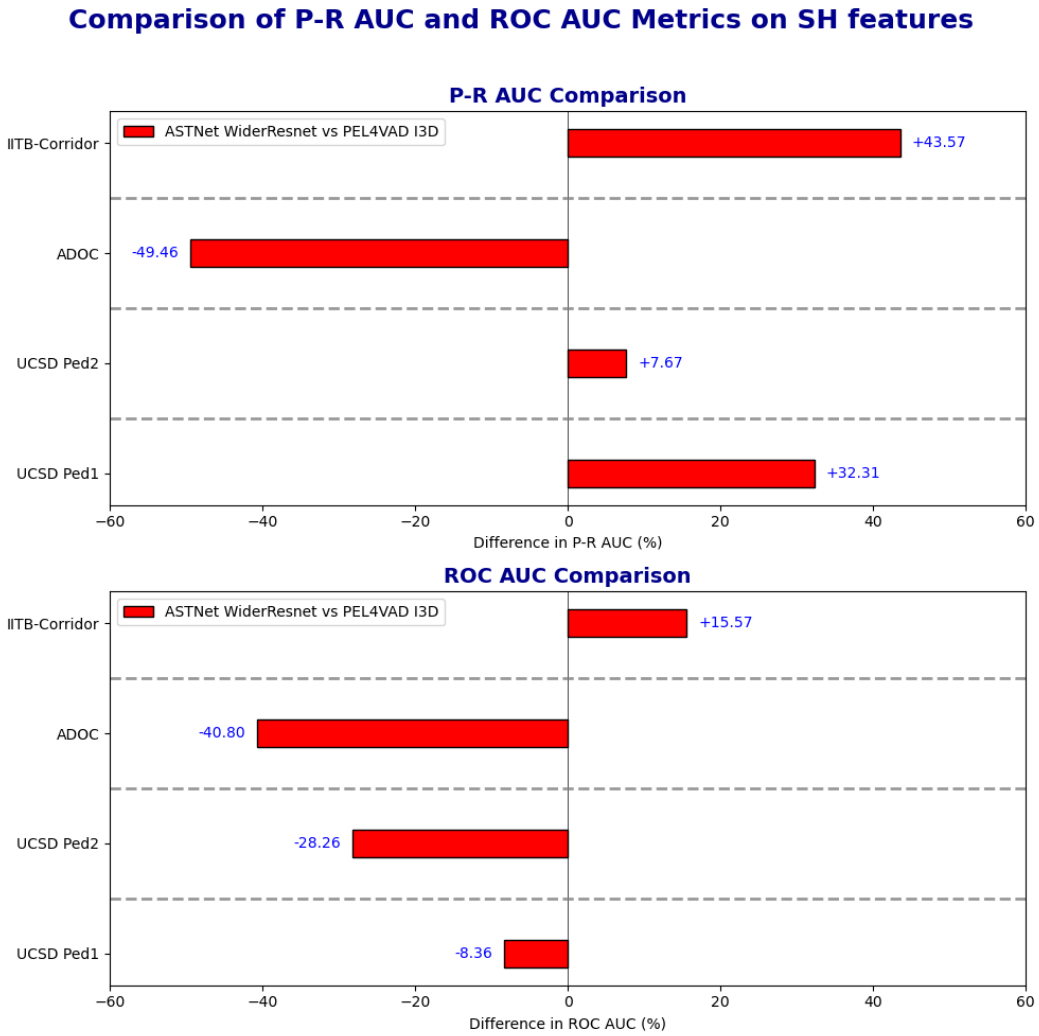


Figure A.3: P-R and ROC result comparison between ASTNet WiderResnet and PEL4VAD I3D. Both models leveraged features extracted from SH Training Set.

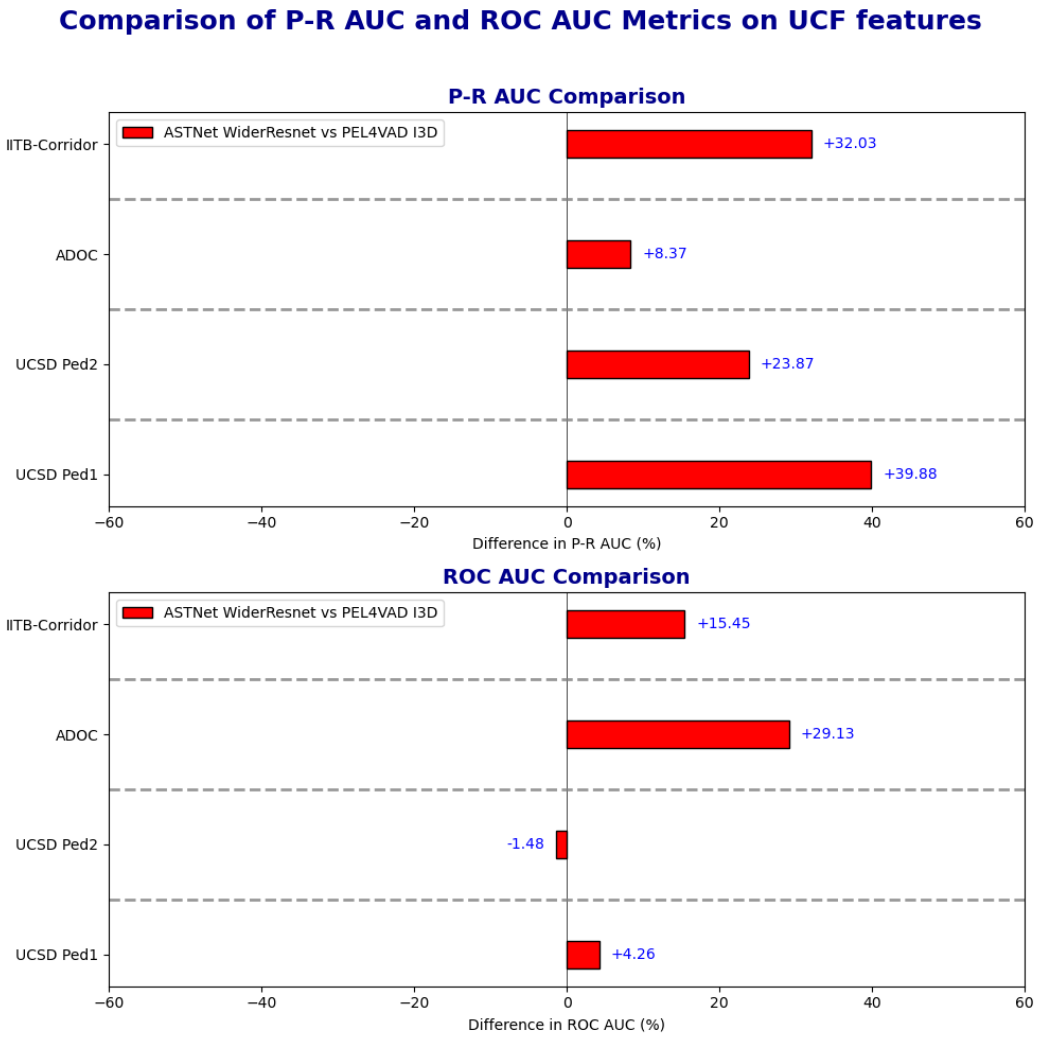


Figure A.4: P-R and ROC result comparison between ASTNet WiderResnet and PEL4VAD I3D. Both models were trained with features extracted from UCF Training Set.

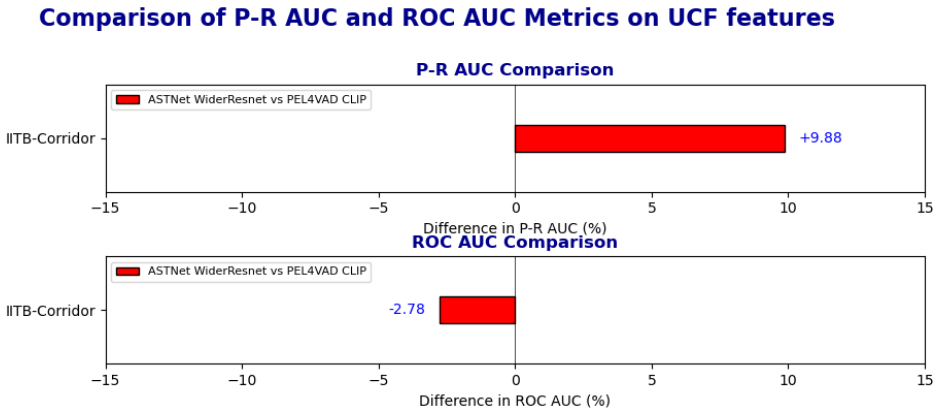


Figure A.5: P-R and ROC result comparison between ASTNet WiderResnet and PEL4VAD CLIP on IITB-Corridor. Both models were trained with features extracted from UCF Training Set.