ANOMALY DETECTION IN VIDEOS BASED ON UNSUPERVISED LEARNING

ANOMALY DETECTION IN VIDEOS BASED ON UNSUPERVISED LEARNING

BY

SHUSHENG LI, M.Sc

A THESIS

SUBMITTED TO THE DEPARTMENT OF COMPUTING AND SOFTWARE

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Doctor of Philosophy (2022) McMaster University

(Computing and Software) Hamilton, Ontario, Canada

TITLE:	Anomaly Detection in Videos Based on Unsupervised Learning
AUTHOR:	Shusheng Li M.Sc. (Computer Technology), Xiamen University, Xiamen, China
SUPERVISOR:	Dr. Wenbo He
NUMBER OF PAGES:	xv, 113

Abstract

Anomaly detection for videos serves an important role in real-world applications. There are two types of anomaly detection for videos: anomalous video detection and anomalous event detection. Anomalous video detection is related to organizing video resources, which enables video-sharing platforms or public sectors to focus on the entire video belonging to the certain category; anomalous event detection in videos is applicable in the scenarios like monitoring in smart homes, smart cities and Internet of Things, which allows surveillance cameras to upload events of interest only so as to suppress the network traffic and reduce storage space in the cloud.

The two types of anomaly detection for videos are challenging. Due to the unavailability of abnormal samples, it is a cumbersome task to train an end-to-end deep supervised learning model. Meanwhile, video data representation is challenging because of the unstructured scheme of video contents. In this paper, we propose different methods for the two tasks. For anomalous video detection, we propose a LSTM-autoencoder-based adversarial learning model (VidAnomaly) without abnormal samples in the training stage. LSTM-autoencoder learns the temporal dependence of the input sequence and reconstructs the input sequence for adversarial learning. In the inference stage, for a given input abnormal sample, the model poorly

reconstructs the sample and the reconstruction error would be high because the proposed model is trained merely on normal samples and its parameters are only suitable for reconstructing normal samples. Based on the high reconstruction error, we detect abnormal samples. For anomalous event detection in videos, we propose Onsite Event Detection (OED), a system that enables real-time event detection on edge. OED first trains a transformer-based autoencoder to learn the spatial-temporal representation of video data observed recently. Then it gains the ability to differentiate eccentric data patterns of events from routine. OED also features an updating strategy that adapts to the changing environment dynamically. As such, OED is capable of continuously detecting events of interest in video streams.

We evaluate our approaches on different datasets in various scenarios (anomalous video detection and anomalous event detection in videos). The experimental results show that our approaches are effective and superior to other methods.

Declaration of Academic Achievement

VideoLoc: Video-based Indoor Localization with Text Information

1. Proposed a video-based indoor localization with text information without the deployment of additional equipment (e.g.,WiFi, Bluetooth, RFID, etc.), which reduces the maintenance cost and is resistant to the strong interference in the complex indoor environment.
2. Demonstrated that VideoLoc achieves high precision of localization and is robust to dynamic environments

CryptoEyes: Privacy Preserving Classification over Encrypted Images

1. Proposed CryptoEyes to address the challenges of privacy-preserving classification over encrypted images.
2. Demonstrated that the proposed two-stream convolutional network architecture for classification over encrypted images captures the contour of encrypted images, therefore significantly boosting the classification accuracy.

VidAnomaly: LSTM-autoencoder-based Adversarial Learning for One-class Video Classification with Multiple Dynamic Images

1. Proposed a LSTM-autoencoder-based adversarial learning system, which preserves spatial and temporal information with multiple dynamic images for videos and achievers unsupervised learning in adversarial scheme.
2. Demonstrated that the system achieves high accuracy and is potential in real-world applications.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor, Dr. Wenbo He, for the continuous support of my Ph.D. study and research, for her patience, motivation, and immense knowledge.

Besides my supervisor, I would like to thank the rest of my supervisory committee members: Dr. Fei Chiang and Dr. Ridha Khedri, for their insightful comments and the encouragement which incentivized me to widen my research from various perspectives.

I also thank my fellow groupmates Yang Bo and Yangdi Lu for the stimulating discussions and all the fun we have had in the past years. Last but not the least, I would like to thank my family for supporting me spiritually throughout my Ph.D. study.

Contents

List of Figures

List of Tables

Notation, Definitions, and Abbreviations

Abbreviations

CNNs	Convolutional Neural Networks
MLP	Multilayer Perceptron
RNNs	Recurrent Neural Networks
GANs	Generative adversarial networks
DNNs	Deep Neural Networks
LSTM	Long Short-term Memory
ViT	Vision Transformer
AE	AutoEncoder

Publications

Publications

Journal Papers

[J2] A segmentation and classification scheme for single tooth in MicroCT images based on 3D level set and k-means++.

Wang L, Li S, *et al.*

Computerized Medical Imaging and Graphics 57 (2017): 19-28.

[J1] An Automatic Segmentation and Classification Framework Based on PCNN Model for Single Tooth in MicroCT Images.

Wang L, Li S, *et al.*

PloS one 11.6 (2016): e0157694.

Conference Papers

[C7] VideoLoc: Video-based Indoor Localization with Text Information.

Li S, He W.

IEEE Conference on Computer Communications (INFOCOM), acceptance rate

of full paper is 19%, May 2021, Vancouver, Canada.

[C6] CryptoEyes: Privacy Preserving Classification over Encrypted Images.
He W, Li S, *et al.*
IEEE Conference on Computer Communications (INFOCOM), acceptance rate of full paper is 19%, May 2021, Vancouver, Canada.

[C5] VidAnomaly: LSTM-Autoencoder-Based Adversarial Learning for One-Class Video Classification With Multiple Dynamic Images.
Li S and He W.
IEEE International Conference on Big Data (Big Data), acceptance rate of full paper is 19%, December 2019, Los Angeles, USA.

[C4] Direct aneurysm volume estimation by multi-view semi-supervised manifold learning.
Wang L, Li S, *et al.*
IEEE 14th International Symposium on Biomedical Imaging (ISBI), April 2017, Melbourne, Australia.

[C3] Volume calculation of CT lung lesions based on Halton low-discrepancy sequences.
Li S, Wang L, *et al.*
Computer-Aided Diagnosis. International Society for Optics and Photonics (SPIE), August 2017, San Diego, USA.

[C2] Direct aneurysm volume estimation by multi-view semi-supervised manifold learning.
Wang L, Li S , *et al.*

IEEE 14th International Symposium on Biomedical Imaging (ISBI), April 2017, Melbourne, Australia.

[C1] Structure Fusion for Automatic Segmentation of Left Atrial Aneurysm Based on Deep Residual Networks.

Wang L, Li S, *et al.*

International Workshop on Machine Learning in Medical Imaging, October 2016, Athens, Greece.

Chapter 1

Introduction

With the advancement of deep generative models (*e.g.,* the variational autoencoder (VAE) (Kingma and Welling, 2014), generative adversarial networks (GANs) (Goodfellow *et al.*, 2014), and Long Short Term memory networks (LSTMs) (Sherstinsky, 2020)), unsupervised representation learning has become an important domain. Anomaly detection is one of the well-known sub-domains of unsupervised learning in the machine learning and data mining. Anomaly detection for videos is challenging because of the high dimensional structure, combined with the non-local temporal variations among frames.

Anomaly detection is an unsupervised learning task in which the aim is to identify abnormal patterns that are infrequent or rare events. In addition, abnormal samples are rarely available to train an end-to-end supervised learning network to separate normal samples from the anomalous ones. This is a complex task since the training data is unbalanced, *i.e.,* class of normal samples comprises frequently occurring objects and regular foreground movements while the anomalous class includes a variety

of unusual events and unseen objects. Due to the fact that abnormal samples are absent during training, poorly sampled or not well defined, the training data containing no anomalies are made available in the training stage. Then, the model is built with the training data representing the normal samples, while detecting anomalous samples in the inference stage. Hence, we resort to unsupervised learning to implement the anomaly detection.

Given a set of training samples without anomalies, our goal is to design a model to learn a feature representation capturing the distribution of normal samples that are by definition frequent or regular events. Any deviations from this normal distribution are identified by measuring the approximation error geometrically in a vector space or the posterior probability of a given model. An approximation error indicates an abnormal pattern or event.

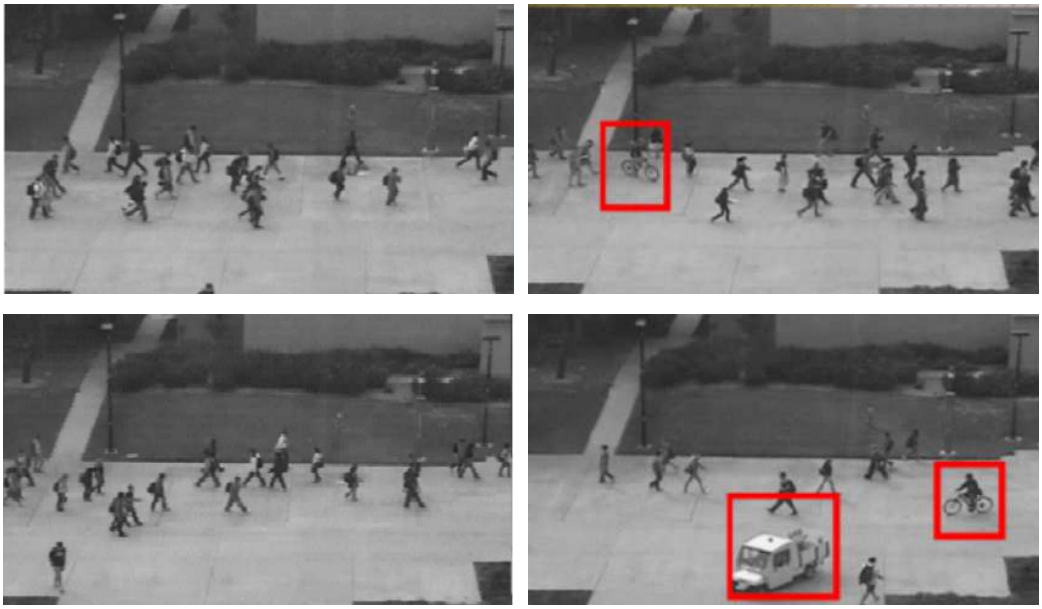
1.1 Background

1.1.1 Anomaly Detection

Anomaly detection is an unsupervised pattern recognition task, which is also defined under different statistical models. The normal behavior is characterized through a number of samples and a statistical model is built with the normal samples that is able to generalize well on unseen samples. The distribution of normal samples is obtained through the training samples $x_i \in X_{train}$, by building a model f_Ω to minimize the reconstruction error over all the training samples:

$$\Omega^* = \arg\min_{\Omega} \sum\nolimits_{x_i \in X_{train}} \|f_\Omega(x_i) - x_i\|^2 \tag{1.1.1}$$

The deviation of the test samples $x_j \in X_{test}$ under this trained model is evaluated as the anomaly score, *i.e.*, $s(x_j) = \|f_{\Omega^*}(x_j) - x_j\|^2$, used as a measure of deviation. The anomalous samples are poorly approximated by the trained model f_{Ω^*} since f_{Ω^*} is trained on normal samples. Anomalous detection is achieved by evaluating a threshold based on the anomaly score. If $s_j > T_{thresh}$, the anomaly is detected. The threshold is a parameter and the variation of the threshold is related to the detection performance.



(a) Normal frames (b) Abnormal frames

Figure 1.1: Samples of normal and abnormal events in videos.

For anomalous event detection in videos, we take the surveillance camera for example. Usually, the background of the videos remains static, while the foreground is moving objects such as pedestrians, traffic, etc. The anomalous video events are those which deviate from routines in appearance and motion patterns. Figure 1.1 demonstrates a few scenes where the vehicles, bicycles, skateboarders, wheelchairs,

etc. are prohibited in an area. We use red boxes to mark these illegal objects. If a frame with any prohibited object, we know that an anomalous event occurs.

1.1.2 Representation Learning for Video Anomaly Detection

Videos are high dimensional data with both spatial-structure and temporal correlations (*i.e.,* spatiotemporal information). A key problem is to find an effective representation of a video clip to capture the features of interest in anomaly detection. Representation learning builds a parameterized model to map an input sample to a lower dimension vector, which is then used to reconstruct the output sample.

Representation learning for reconstruction. There are many methods to represent the different linear and non-linear transformations, such as Principal component analysis (PCA) and Autoencoders (AEs). They model the normal events in surveillance videos and abnormal samples represent any deviations that are poorly reconstructed.

Representation learning for predicting the frame. A video clip consists of a sequence of frames and the frames are temporal correlated, so it is important to take the advantages of temporal information among the frames. The goal of reconstruction is to learn a generative model that successfully reconstructs video frames, while predictive model is to predict the current frame or its encoded representation using the past frames. One example is the convolutional LSTM model.

Generative models. Generative models include Variational Autoencoders (VAE), Generative Adversarial Networks (GANs) and Adversarially trained AutoEncoders (AAE). They are able to model the likelihood of normal samples in videos in an end-to-end deep learning framework without abnormal samples in the training stage. An

important common aspect in all these models is representation learning. This means that the feature extraction of input training data is a key component for the task of anomaly detection.

1.1.3 Reconstruction Models

Given an input training video $X_{train} \in R^{N*d}$, N is the number of frames and $d = r * c$ denotes the number of pixels per frame. d is also the dimensionality of each vector corresponding to the frame. Reconstruction models focus on reducing the reconstruction error for normal samples, including the Principal Component Analysis (PCA), Convolutional AutoEncoder (ConvAE), and Contractive AutoEncoders (CtractAE).

Principal Component Analysis. The goal of PCA is to find the directions of maximal variance in the training data. In the case of videos, PCA is to model the spatial correlation for pixel values that are components of the vector representing a frame at a particular time instant. A set of orthogonal projections de-correlate the features in the training set using input training matrix X that has zero mean: $\min_{W^TW=I} \left\|X - (XW)W^T\right\|_F^2$, where $W^TW = I$ represents an orthonormal reconstruction of the input X. The projection XW is a vector in a lower dimensional space. Each frame is associated with continual optical flow magnitude, learning atomic motion patterns with standard PCA on the training set, and evaluating reconstruction error on the test optical flow magnitude.

Autoencoders. An autoencoder is a type of artificial neural network used to learn efficient codings of unlabeled data (unsupervised learning) and trained by back-propagation. It provides an alternative to PCA for dimensionality reduction by reducing the reconstruction error on the training set as shown in Figure 1.2.

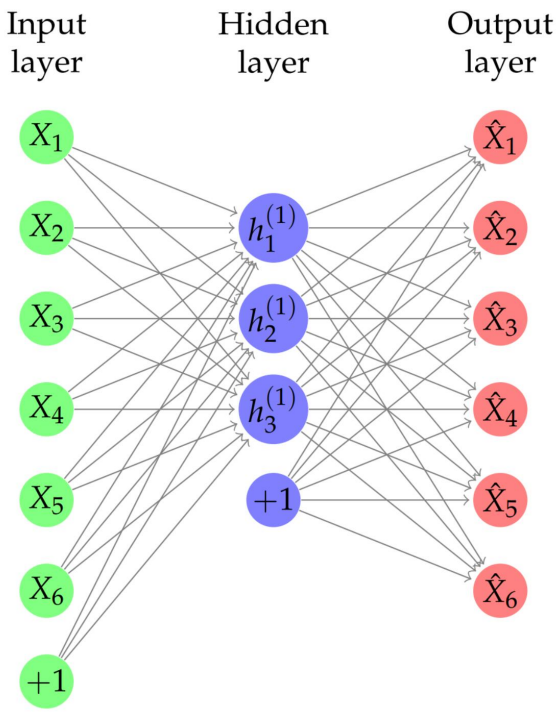


Figure 1.2: Autoencoder with three layers: input layer, output layer and single hidden layer.

It takes an input in the input layer and maps the input to the latent space representation in the hidden layer. The autoencoder performs a non-linear point-wise transform of the input with activation functions, such as rectified linear unit (ReLU) and sigmoid. The reconstruction error is obtained by calculating the deviation between the input and the output.

1.2 Objective and Scope

Upon the study of anomaly detection for videos, we make several observations that heavily affect the detection accuracy. First of all, although the deep supervised learning models show good performance in computer vision tasks, these models are only

applicable when the events of interest (*i.e.,* positive events) are clearly defined and training samples are correctly labeled with appropriate percentage of positive events. Therefore, it is a cumbersome task to train an end-to-end deep supervised learning model if without high quality labeled samples. Second, autoencoder is an unsupervised learning model, so it doesn't require abnormal samples for training and is able to capture the distribution of normal samples in the training stage. The architecture based on autoencoder is beneficial for achieving unsupervised learning. Third, the long-range temporal information is important for anomaly detection for videos. Extending the temporal dimension of 3D CNNs to accommodate longer video clips (Varol et al., 2018) or using sparse selection approaches (Wang et al., 2016; Lan et al., 2017; Zolfaghari et al., 2018) are both effective ways to enable networks to study the long-range temporal information. However, both fail to preserve the temporal information of the entire video since the input size (length of video clips) is limited. Fourth, current anomaly detection approaches mainly focus on the accuracy using high-performance computers. In practice, we prefer to filter out irrelevant contents and only transmit the events of interests on the edge computing devices.

Given the above observations and issues, the objective of this research is to explore and develop new approaches that achieve anomaly detection for videos in unsupervised learning manner. The aim is to reduce the dependency on large manually annotated collections of videos, helping the research in anomaly detection become more feasible for those with limited computing resources. To develop new approaches, the following questions should be answered.

1. Since long-range temporal information is important for videos, it is difficult to preserve the temporal information of the entire video for anomalous video detection.

Simple 1D/3D convolution fails to fully preserve the temporal information since they only accept the fixed-size input. CNN+RNN model could accept the inputs with various sizes, but it would severely overfit. Would it be possible to use the compact representation to encode the temporal evolution of the frames of the video, thereby capturing the dynamics from the whole video?

2. An reason to make the representation and modeling of video data very challenging is that anomalies are highly contextual and hard to be well defined. It leads to the unavailability of abnormal samples. therefore, we cannot rely on supervised learning to catch the abnormal events. Is it possible to train an effective model to represent the normal samples in the training stage without abnormalities and then use the trained model to detect anomalies in the inference stage? Then, based on the high reconstruction error and the big difference for given samples, abnormal samples will be detected.

3. With the development of surveillance systems, millions of video cameras have been deployed for security and safety purposes. These always-on cameras continuously record large volumes of video data. However, large segments of videos only contain background or irrelevant contents. Would it be possible to achieve real-time to detect abnormal events therefore suppressing the network traffic and reducing storage space based on the proposed method?

1.3 Contributions

The main contribution of this thesis shall be the investigation of two types of anomaly detection for videos. We focus on anomalous video detection and anomalous event

detection in videos. More specifically, in this work we make the following contributions:

1. We propose a LSTM-autoencoder-based adversarial learning model for anomalous video detection without abnormal samples in the training stage. To represent video data with temporal and spatial information, we adopt multiple dynamic images which contain the temporal evolution of video frames and represent the video contents at the level of the image pixels. Multiple dynamic images are viewed as the input sequence with temporal and spatial information and achieve dimension reduction of original video data. In the inference stage, if a given abnormal sample is fed to the model, the reconstruction error will be high because the proposed model is trained merely on normal samples and its parameters are only suitable for reconstructing normal samples. Based on the high construction error, we will be able to detect abnormal samples.

2. We propose Onsite Event Detection (OED), a system that enables real-time event detection on edge. OED first trains a transformer-based autoencoder in adversarial learning to learn the spatial-temporal representation of video data observed recently. Then it gains the ability to differentiate eccentric data patterns of events from routine. OED also features an updating strategy that adapts to the changing environment dynamically. As such, OED is capable of continuously detecting events in video streams.

3. We achieve an unsupervised learning manner in adversarial training for both tasks. There are two networks in the proposed architecture reinforcing each other, and after training the model reconstructs the normal samples more accurately with a small reconstruction error.

1.4 Outline

This thesis is organized as follows. We present the proposed method for anomalous video detection, as well as the experiment results in chapter 2. The proposed vision transformer based framework for anomalous event detection in videos is described in chapter 3. We conclude the thesis in chapter 4.

Chapter 2

VidAnomaly: LSTM-autoencoder-based Adversarial Learning for Anomalous Video Detection with Multiple Dynamic Images

2.1 Citation and Main Contributor

Li, Shusheng, and Wenbo He. "VidAnomaly: LSTM-Autoencoder-Based Adversarial Learning for One-Class Video Classification With Multiple Dynamic Images." 2019 IEEE International Conference on Big Data (Big Data), acceptance rate of full paper is 19%, December 2019, Los Angeles, USA.

The main contributor to this paper is the first author - Shusheng Li (contributes more than 80%).

2.2 Copyright

2.3 Abstract

Anomalous video detection serves an important role when abnormal videos are absent during training, poorly sampled or not well defined. However, anomalous video detection is challenging. Due to the unavailability of abnormal samples, it is a cumbersome task to train an end-to-end deep supervised learning model. Meanwhile, video data representation is challenging because of the unstructured scheme of video contents. To represent video data with temporal and spatial information, we propose multiple dynamic images in our task because dynamic image encodes the temporal evolution of video frames and represents video contents at the level of the image pixels. Multiple dynamic images are viewed as the input sequence with temporal and spatial information and achieve dimension reduction of original video data.

In this paper, we propose a LSTM-autoencoder-based adversarial learning model for anomalous video detection ("VidAnomaly") without abnormal samples in the training stage. Our architecture is composed of three sub-networks. LSTM-autoencoder network (R) learns the temporal dependence of the input sequence and reconstructs the input sequence for the discriminator network (D) to achieve adversarial learning. The novelty of the proposed model is that we add an additional LSTM-encoder

network (A) to obtain the latent representation of the reconstructed sequence. Minimizing the distance between the two latent representations from R and A benefits the model to further capture the training data distribution because it forces the LSTM-autoencoder network to yield an essential representation of training samples in latent space. In the inference stage, for a given abnormal sample as the input, the model poorly reconstructs the input abnormal sample and the reconstruction error would be high because the proposed model is trained merely on normal samples and its parameters are only suitable for reconstructing normal samples. Based on the high reconstruction error, we detect abnormal samples. The experimental results show that VidAnomaly learns the distribution of normal samples effectively and is superior to other methods.

2.4 Introduction

In recent years videos have become one of the most dominant forms of big data on the Internet. This is achieved by rapid advances in multimedia technologies and the proliferation of online video hosting and sharing services (Chen *et al.*, 2016) such as Youtube, Vimeo, Metacafe, etc. According to the Cisco Visual Networking Index (Barnett *et al.*, 2018), IP video traffic will account for 82% of all IP traffic by 2022, up from 75% in 2017.

The explosion of video is accompanied with how to manage and organize the tremendous video resources. Anomalous video detection serves an important role in organizing video resources. Anomalous video detection can be illustrated in the context of one-class video detection (Gardner *et al.*, 2006; Khan and Madden, 2014), aiming to establish the models to classify the normal samples from the abnormal ones

when the abnormal class is absent, poorly sampled or not well-defined (Sabokrou *et al.*, 2018). Anomalous video detection is suitable in the following two scenarios:

- Abnormal class is complex. Anomaly detection is different from the classification task in which there is a determined number of category in the classification task. For anomaly detection, the samples that differ from the normal ones are considered as the abnormalities, so that there are too many abnormal cases and the number of the category is unpredictable in advance. Also, it is not possible to label all types of anomalous class. This limits the typical classification approaches for anomaly detection. For example, video-sharing platforms focus on the specific video contents such as fire disaster, wedding ceremony, etc., when these kinds of video contents become the issue in the news because heavily recommending the hot video contents obtains a high click-through rate. However, there are too many abnormal cases and the categories of video contents can be up to thousands of classes (Lee *et al.*, 2018). It is expensive and cumbersome to label the video data accurately and labeling all types of abnormal videos is impossible. The task of anomalous video detection is to efficiently model such problems.

- Abnormal samples are insufficient. The typical classification requires extensive supervised training using large amounts of annotated data. However, for anomaly detection, abnormal samples are insufficient for training because they are difficult to be collected, while normal samples are easy to be obtained and sufficient. For example, in areas like ATM room, server room, etc., there may be security problems such as robbery, destroying the machines, etc. The abnormal cases are not common but pose a security threat. On the other hand, normal

cases are not difficult to be collected. Due to the insufficient size of abnormal samples, training an end-to-end deep supervised learning model is cumbersome.

The above-mentioned examples show that anomalous video detection provides the service for video-sharing platforms and benefits video analytics in government, public sector and society in general.

In this paper, we focus on anomalous video detection when abnormal videos are complex or insufficient. However, anomalous video detection is challenging. Compared with other forms of multimedia such as audio, images, etc., videos are more information-abundant and complicated. Video data provide both temporal and spatial information, which is beneficial for the description and analysis of video contents. Meanwhile, video data belong to unstructured data so that the information either does not have a pre-defined data model or is not organized in a pre-defined manner. This results in irregularities, making it difficult in organization, understanding, classification, and retrieve. Nevertheless, understanding video contents is the first step of anomalous video detection.

To efficiently and accurately understand the contents of videos in computer vision, video feature representation is a critical process, which maps various video contents to digital contents. Much effort has been made on video representation. Recently, the learnable representations such as deep convolutional neural networks (CNNs) have achieved good performance in many video understanding tasks. As the frames in the video can be viewed as a set of independent and static images, the approaches (Krizhevsky *et al.*, 2012) packed a short sequence of video into an array of images and then fed them into CNNs. Another improvement (Ji *et al.*, 2012) is extending CNNs to replace 2D filters with 3D ones, which extracts temporal features. However,

a downside of CNN-based video representation methods is that they only capture the local changes within a small time window and fail to capture long-term dynamic patterns associated with a certain of video content. Besides, in our task, the correlation among consecutive video frames (temporal information) is significant, which further strengthens the understanding of video contents. CNN-based video representation that only captures temporal features in a small time window is not suitable for our task.

The above-mentioned limitations are common in CNN-based feature representation. To address these issues, we introduce *dynamic image* (Bilen *et al.*, 2016) in our task. Dynamic image is a novel compact representation of the video. This compact representation is able to encode the temporal evolution of the frames of the video, thereby capturing the dynamics from the whole video. Dynamic image represents the video content at the level of image pixels so that the spatial information is also preserved. Profiling the whole video based on visual characteristic instead of intermediate feature vector is helpful for the follow-up processes in our framework because visual characteristic is more appropriate to be processed by deep learning models. Another advantage is that the extraction of dynamic image is simple and efficient and it makes the original video data to achieve dimension reduction for reducing the redundance.

However, summarizing the video contents in a single image may result in only a few samples for training. The model for anomalous video detection also requires sufficient normal samples for training. Therefore, we propose video data augmentation in our task. In addition to increasing the number of training samples, video data augmentation aids in reducing the effects of overfitting (Karpathy *et al.*, 2014) and

being more robust against noise (Sabokrou *et al.*, 2018). We propose two simple and efficient methods for video data augmentation: a) Directly adding a Gaussian noise to the existing training samples. This corrupts the training samples and enables our proposed anomalous video detection model to be more robust to noise. b) Spatial augmentation of training samples. This is the same as images (Krizhevsky *et al.*, 2012), resampling the video frames and flipping them randomly.

Summarizing an entire video sequence into a dynamic image may suffer from temporal and spatial information loss, especially for a very long video clip. For a very long video clip, the whole video sequence summarized in a single dynamic image is difficult and plenty of temporal and spatial information may be neglected. To address this limitation, we propose to extract multiple dynamic images from a given video sequence. In detail, we segregate the given video into segments and then extract the dynamic images from every segment. Multiple dynamic images provide a fixed number of dynamic images for each video, which also benefits deep learning models that process sequences of data (containing temporal information) like long short-term memory (LSTM).

In recent years, anomaly detection has made significant progress. The common methods for anomaly detection are based on proximity, distance, or nearest neighbor. For example, the previous research focalizes on distance-based approaches for anomaly detection using K nearest neighbor distance-based methods (Campos *et al.*, 2016).

However, as the number of dimensionality increases, especially for even ultrahigh-dimensional data, proximity, distance, and nearest neighbor become less meaningful (Kriegel *et al.*, 2008). The distance-based methods also increase the computational burden for anomaly detection in high dimensional data. Hence, much previous work has focused on dimensionality reduction based anomaly detection, such as PCA based multivariate scheme (Camacho *et al.*, 2016), non-linear dimensionality reduction based autoencoder (Sakurada and Yairi, 2014), etc. Nevertheless, these approaches fail to keep the essentiality of samples because they only use simple distributions like isotropic Gaussian distribution, resulting in poor performance on anomaly detection.

In this paper, we propose adversarial learning system for anomalous video detection by capturing essential features of high dimensional training data in the low-dimension latent space. Adversarial learning like generative adversarial networks (GANs) has the ability to efficiently learn the training data distribution and preserve the essentiality of the data. This is because the architecture of adversarial learning allows it to directly model the real-world unstructured data based on a data-driven scheme in adversarial mechanism.

The training samples in our task are multiple dynamic images that are the sequences. Therefore, unlike most of the adversarial learning architectures using CNN-like structure (Akcay *et al.*, 2018), we propose LSTM-autoencoder-based adversarial learning to process the sequences of the input data, which captures the training data distribution effectively and further preserves the temporal and spatial information of the input sequence. Different from the work (Sabokrou *et al.*, 2018) where two

networks of GAN are used for anomalous video detection, our proposed architecture is composed of three-subnetworks and the novelty is that we add the additional LSTM-encoder network to obtain the latent representation of the reconstructed sequence. Minimizing the distance between the two latent representations from LSTM-autoencoder network and the additional LSTM-encoder network aids in capturing the training data distribution and detecting abnormalities during the inference stage.

In this paper, we propose LSTM-autoencoder-based adversarial learning for anomalous video detection with multiple dynamic images ("VidAnomaly"). VidAnomaly proceeds by extracting multiple dynamic images from the training samples and then training the LSTM-autoencoder-based adversarial learning model. This model is trained only on normal samples that belong to the target class and then its parameters are not suitable for reconstructing abnormal samples. In the inference phase, when the input is an abnormal sample, the proposed model fails to reconstruct the abnormal sample accurately, because the parameters in this model are only suitable for reconstructing normal samples. Based on the high reconstruction error and the big difference between the two latent representations, we can distinguish the abnormal samples.

Our system "VidAnomaly" is designed to tackle the main challenges of anomalous video detection and has the following advantages. First, we propose to use video data augmentation, which does not only increase the number of video data but also allows our proposed method to be more robust to noise and avoid overfitting. Second, we propose to extract multiple dynamic images from a given video to profile the video with temporal and spatial information. Dynamic image is characterized by the temporal evolution of the frames of the video. However, summarizing the whole video

into a single dynamic image may result in critical information loss, especially for a very long video clip. Therefore, we break the video into segments and then extract multiple dynamic images from each segment. Third, we propose LSTM-autoencoder-based adversarial learning to encode and decode the training samples based on the data-driven scheme. This further aids in preserving temporal information and yielding a low-dimension latent representation of the input data. Fourth, the novelty of the proposed model is that in addition to minimizing the distance between the input samples and the reconstructed samples in original space, we minimize the distance between the two low-dimension latent representation from LSTM-autoencoder network and the additional LSTM-encoder network in latent space. The additional LSTM-encoder network helps to learn to encode the normal samples during the training phase. This has the best possible representation of normal samples that could lead to their reconstruction. Fifth, we propose adversarial learning model for anomalous video detection without abnormal samples for training. In the training phase, the proposed adversarial model is trained merely on the normal samples and captures the distribution of normal samples, so that its parameters are only suitable for reconstructing the normal samples. In the inference phase, for the abnormal sample as the input, the model is confused and unable to reconstruct the input accurately. Therefore, the reconstruction of the input is very different from the original input. In this way, the abnormal sample is detected.

The rest of this chapter is organized as follows. A brief review on related work is given in Section 2.5. Section 2.6 introduces preliminaries for dynamic image, LSTM, and GAN. The proposed system is described in Section 2.7. Section 2.8 presents the experimental results. Section 2.9 concludes the chapter.

2.5 Related work

One-class video classification via adversarial learning is closely related to video representation, anomaly/outlier detection, self-representation, and adversarial learning.

Video representation. Video representation is one of the most important steps in video classification, retrieve, and understanding. In the literature, several methods have been proposed. Traditionally, encoding frames according to low-level features, such as HOG (Dalal and Triggs, 2005), SIFT (Lowe, 2004), and so on. However, these feature descriptors often suffer from appearance variations of dynamic scenes and objects (Chang *et al.*, 2017). Deep CNNs have improved dramatically the performance in representing videos and can be broken into two categories. The first category considers the video as a stream of individual frames (Yue-Hei Ng *et al.*, 2015) or as a short and smooth transition among frames (Simonyan and Zisserman, 2014). The other category is to replace 2D filters with 3D ones by extending CNNs (i.e. 3D CNNs) (Tran *et al.*, 2015; Simonyan and Zisserman, 2014; Gkioxari and Malik, 2015), which extracts the temporal information within a small window. However, the disadvantage of these CNN-based video representations are unaware of the dynamics of the video and fail to capture long-term dynamic patterns associated with a certain of video content. Therefore, we propose to use dynamic image (Bilen *et al.*, 2016) to encode the temporal evolution of the frames in the video.

Anomaly/outlier detection. Anomaly or outlier detection has received tremendous attention because of numerous practical applications. We focalize on the most related work that applies machine learning models for anomaly detection. The author in (Gaddam *et al.*, 2007) utilized ID3 decision tree learning method to detect abnormal samples in computer networks. Abe *et al.* (Abe *et al.*, 2006) viewed the

anomaly detection problem as a classification problem and proposed supervised active learning for such a problem. Ashfaq *et al.* (Ashfaq *et al.*, 2017) presented a fuzziness based semi-supervised learning by for intrusion detection system. However, due to the insufficient sizes or unavailability of abnormal data, training an end-to-end machine learning network is not straightforward and effective. In this work, we propose adversarial learning-based anomaly detection framework, the advantage of which is that there is no need for abnormal samples in training. Meanwhile, it is able to distinguish between normal and abnormal samples in the inference phase.

Self-representation. It has been shown that self-representation (e.g. auto-encoder) is a useful tool for anomaly detection (Xu *et al.*, 2015; Sabokrou *et al.*, 2016). It is assumed that the input samples are reconstructed using the samples from the target class. If the reconstruction error is high, it indicates that the testing sample is more probably abnormal. In our task, the training samples are multiple dynamic images with the temporal information, so we propose to use LSTM-autoencoder-based architecture to reconstruct the sequence of the input.

Adversarial learning. The most representative adversarial learning method is generative adversarial networks (GANs) (Goodfellow *et al.*, 2014) in recent years, because it has shown huge success in generating data. GANs has also been widely used to capture the potential latent representation of the training data based on auto-encoder architecture (Chang *et al.*, 2017; Ghasedi *et al.*, 2019). GANs is based on a two-player game between two networks, i.e., a discriminator and a generator, introduced by Goodfellow and co-authors in (Goodfellow *et al.*, 2014). The generator is to generate the real-world data (e.g. images) and is trained to fool the discriminator as much as possible. The discriminator takes true data and generated data and tries

to discriminate true data from the generated data as well as possible. In order to generate the data similar to true data, the generator has to capture the training data distribution effectively and accurately so that it is able to generate the data almost the same as true data. In this process, GANs obtains a low-dimension vector (representation) to characterize the original training data. In our task, we propose GAN-like architecture for anomalous video detection, making full use of the ability to generate a low-dimension latent representation.

2.6 Preliminaries

2.6.1 Dynamic Image

Dynamic image is a novel compact representation of videos. The main advantage of dynamic image is to encode the temporal evolution of the frames of the video. Videos comprise a large majority of the visual data with temporal and spatial information. Dynamic image is able to summarize the video content in a single image.

Long-term dynamics and temporal patterns are very crucial cues for the recognition of actions (Fernando *et al.*, 2016a, 2015, 2016b; Hoai and Zisserman, 2014; Ryoo *et al.*, 2015). However, it is challenging to represent complex long term dynamics, particularly compact representations that can be processed efficiently. Several effective representations of long-term dynamics have been obtained through temporal pooling of frame features in a video. Dynamic image is a new long-term pooling operator and it is simple, efficient, yet powerful for compact representation of videos.

Some examples of dynamic images are shown in Figure 2.1. We can observe that

Figure 2.1: A few examples of dynamic images that summarize the appearance and dynamics of videos, encoding the temporal evolution of the frames. The examples are from (Bilen *et al.*, 2016)

the dynamic images focus mainly on the acting objects. On the other hand, background pixels tend to be averaged away. Hence, the pixels in the dynamic image focus on the motion of the obvious contents in videos. This means that they may contain the information necessary to video understanding. In addition, dynamic images are different for actions of different speeds. Dynamic images are similar to some other imaging effects that convey motion and time, such as motion blur or panning as shown in Figure 2.4. Dynamic images capture the time and motion (temporal information) by integrating and reordering the pixel values over time.

The first step of constructing dynamic image is to represent a video as a ranking function for its frames $I_1, I_2, ..., I_T$, where T is the length of the video. the novelty of (Bilen *et al.*, 2016) is that $\Psi(I_t)$ is directly the RGB image pixels, instead of the local features (e.g. HOG, SIFT, MBH) extracted from every video frame. Although this idea is simple, the final representation is the format of a single still image because of the idea. In detail, $\Psi(I_t)$ denotes an operator that stacks the RGB component of every pixel for video frame I_t on a large vector. $\Psi(I_t)$ can be also incorporate a simple non-linear kernel, such as the square root function. The time average of these video frames up to time t can be written as follows:

$$A_t = \frac{1}{t} \sum_{i=1}^{t} \Psi(I_i) \tag{2.6.1}$$

The ranking function for its frames associates to each time t a score $S(t|P) = \langle P, A_t \rangle$, where P is a vector of parameters. The scores reflect the rank of the frames in the given video. To keep the order, later video frames have larger scores, i.e., $q > t \Rightarrow S(q|P) > S(t|P)$. The parameters P is obtained by solving a convex optimization

problem based RankSVM (Smola and Schölkopf, 2004) formulation:

$$F(P) = \frac{2}{T(T-1)} \sum_{q>t} \max\{0, 1 - S(q|P) + S(t|P)\} + \frac{\lambda}{2} \|P\|^2 \tag{2.6.2}$$

$$P^* = \rho(I_1, I_2, ..., I_T; \Psi) = \arg\min_P F(P) \tag{2.6.3}$$

where first term in the objective function (2.6.2) is a hinge-loss which counts how many pairs $q > t$ are incorrectly ranked by the scoring function. The second term of (2.6.2) is the quadratic regularize for SVMs. Equation (2.6.3) defines a function $\rho(I_1, I_2, ..., I_T; \Psi)$ that represents a video to a single vector of parameters P^*. The most important advantage of the vector is that it has the same number of elements as an individual video frame, which means that the final representation of the given video can be transformed to the format of a 2D image. In this way, we directly adopt deep learning models at the level of the image pixels instead of intermediate feature representation.

2.6.2 Long Short Term Memory

2D or 3D CNNs fail to capture enough temporal information because they only capture the local changes within a small time window which is not meaningful for videos with longer-term patterns. Video data have a more abundant and rich source of visual information compared with images, i.e., both temporal and spatial information in the video. Therefore, learning a good representation with temporal gains a better understanding of the content of videos. Long Short Term Memory (LSTM) achieves good performance in recent years to represent video sequences since the architecture

of LSTM consists of a forget gate, input and output gates, and a memory cell. LSTM operates as follows. The first step in LSTM is to decide what information is forgotten from the cell state. A sigmoid function in the "forget gate" makes the decision, which has two external sources, i.e., the current frame x_t and the previous hidden states h_{t-1}. The output of the "forget gate" is a number with the range between 0 and 1 associated with the previous cell state c_{t-1}, where 1 represents "completely keep the cell state", while 0 represents "completely remove it". The "forget gate" is formalized as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.6.4}$$

The next step is to update the input information, called "input gate" that receives inputs:

$$I_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.6.5}$$

Then, the old cell state C_{t-1} is updated into the new cell state C_t. This step is associated with the "forget gate" and the tanh non-linearity as follows:

$$C_t = f_t * C_{t-1} + I_t * \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{2.6.6}$$

Finally, "output gate" is based on the cell state. The final output is obtained by multiplying the activation and is updated by:

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{2.6.7}$$

$$h_t = O_t * \tanh(C_t) \tag{2.6.8}$$

The key advantage of LSTM over recurrent neural networks (RNN) is that the cell state is optimized in the forget gate where the unimportant information is removed. In this work, we propose to use LSTM autoencoder to reconstruct the sequences of the input, thus obtaining the effective latent representation and reconstructed data for adversarial learning. The detail is presented in Section 2.7.

2.6.3 GANs for anomaly detection

GANs have shown outstanding success in the field of generating data introduced in (Goodfellow *et al.*, 2014). The standard GAN is formulated as a two-player game in the adversarial network, which is composed of two networks, i.e., a generator G and a discriminator D. The generator takes a random input of z and $p(z)$ is the distribution of z. The aim of the generator is to generate realistic data (e.g. images) that tries to follow the same distribution as the true data $p(x)$. The discriminator takes true and generated data and tries to discriminate them as well as possible, while the generator is trained to fool the discriminator as much as possible. The two networks are learned in a two-player min-max game. Hence, the loss function of GANs is formulated as:

$$\min_G \max_D (\mathbb{E}_{x\sim p(x)}[\log D(x)] + \mathbb{E}_{z\sim p(z)}[\log(1 - D(G(z)))]) \tag{2.6.9}$$

The goal of the generator is to fool the discriminator, while the goal of the discriminator is to distinguish between true and generated data. Therefore, the generator is forced to learn the distribution of true data, while the discriminator finds the

boundaries between the true and generated data distribution.

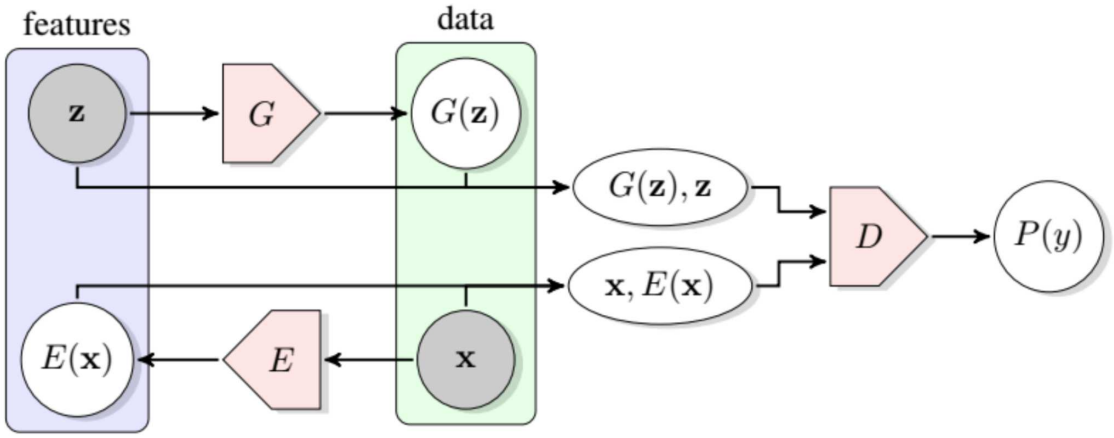


Figure 2.2: Bidirectional GAN (BiGAN) (Donahue *et al.*, 2016) extends the GAN framework.

Bidirectional GAN (Donahue *et al.*, 2016) extends the GAN framework as shown in Figure 2.2, which includes an encoder that learns the inverse of the generator. The Bidirectional GAN (BiGAN) learns a mapping simultaneously from latent space to data and vice versa in the training stage. The encoder is a non-linear function and the discriminator learns to classify real and fake samples.

Anomaly detection using GANs is an attractive research field. Schlegl *et al.* (Schlegl *et al.*, 2017) were the first to propose such a concept. They propose AnoGAN based on a standard GAN, with training only on normal samples to learn a mapping from the latent space representation to the realistic sample. After that, this learned representation maps unseen samples back to the latent space. Training on normal samples only makes the generator to learn the distribution of normal samples. After training, the model learns how to generate normal samples. When an abnormal sample is fed into the model, the model poorly reconstructs the sample. Hence, the difference between the input samples and the reconstructed ones will highlight the

anomalies. The two steps of training and identifying anomalies are illustrated in Figure 2.3.

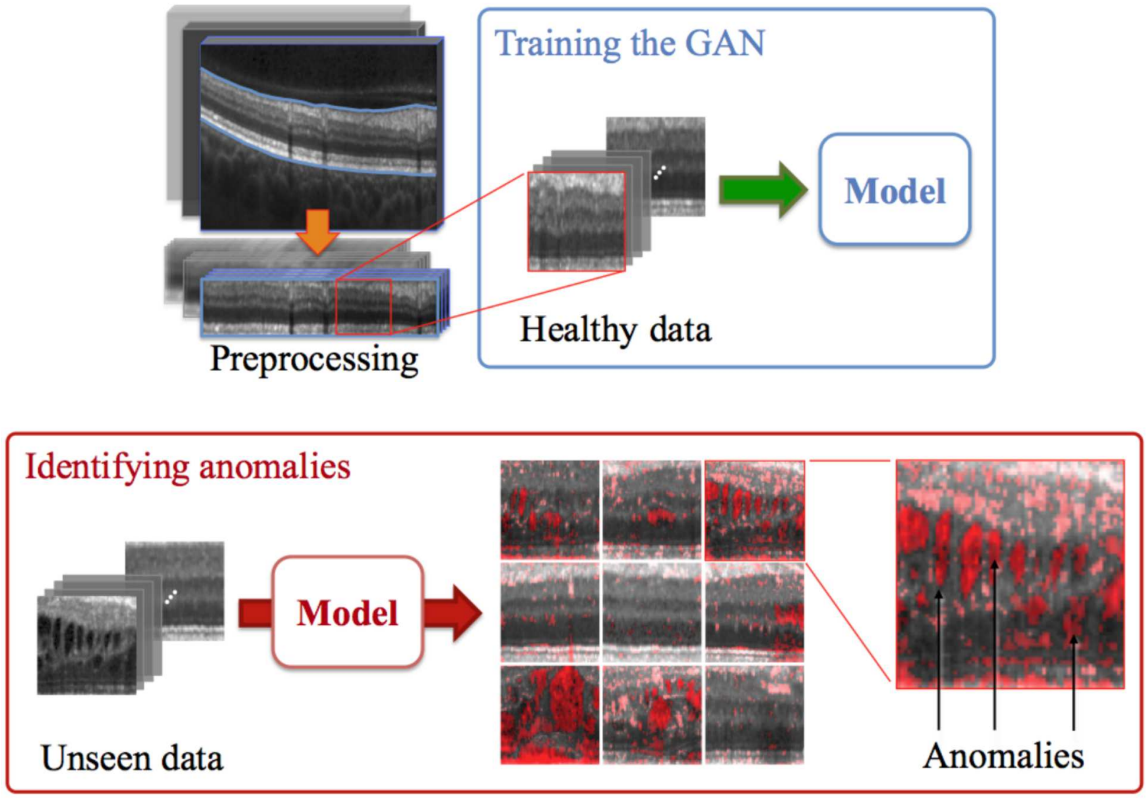


Figure 2.3: AnoGAN (Schlegl *et al.*, 2017). The model is trained on normal samples. The reconstructed image is used to identify anomalies.

The contributions of AnoGAN are showing that GANs can be used for anomaly detection and introducing a new mapping scheme from latent space to input sample space. However, there are also disadvantages. It requires optimization steps for each new input and the objective does not take into account the inverse mapping learning.

To improve performance of AnoGAN, Zenati *et al.* proposed (Zenati *et al.*, 2018) a BiGAN-based method. EGBAD (Efficient GAN-Based Anomaly Detection) outperforms AnoGAN execution time, bringing the BiGAN structure to the anomaly

detection. EGBAD tries to solve the disadvantages of AnoGAN by learning an encoder to map input samples to their latent representation during the training. The main contribution of the EGBAD is to compute the anomaly score without optimization steps for each new input during the inference.

Our proposed GAN-based anomaly detection consists of three networks, a discriminator, a generator, and an additional encoder. Besides, instead of mapping the random input z into the data sample, we propose to use LSTM-autoencoder to encode and decode the input data, obtaining the reconstructed data for the discriminator network to achieve adversarial learning. The novelty is that we add an additional LSTM-encoder to further aids in capturing the distribution of training data. We present the detail in Section 2.7.

2.7 System Design of "VidAnomly"

In this section, we first present the overview of "VidAnomly" in Section 2.7.1. Next, details about training for LSTM-autoencoder-based adversarial network are given in Section 2.7.2. Section 2.7.3 explains how anomalous video detection works.

2.7.1 The overview of "VidAnomly"

"VidAnomaly" system includes three parts: 1) preprocessing module, 2) training module and 3) anomalous video detection module.

Preprocessing module. As summarizing the video content in dynamic images leads to not enough videos available for training, we propose video data augmentation to increase the number of training samples. This also reduces the effects of overfitting

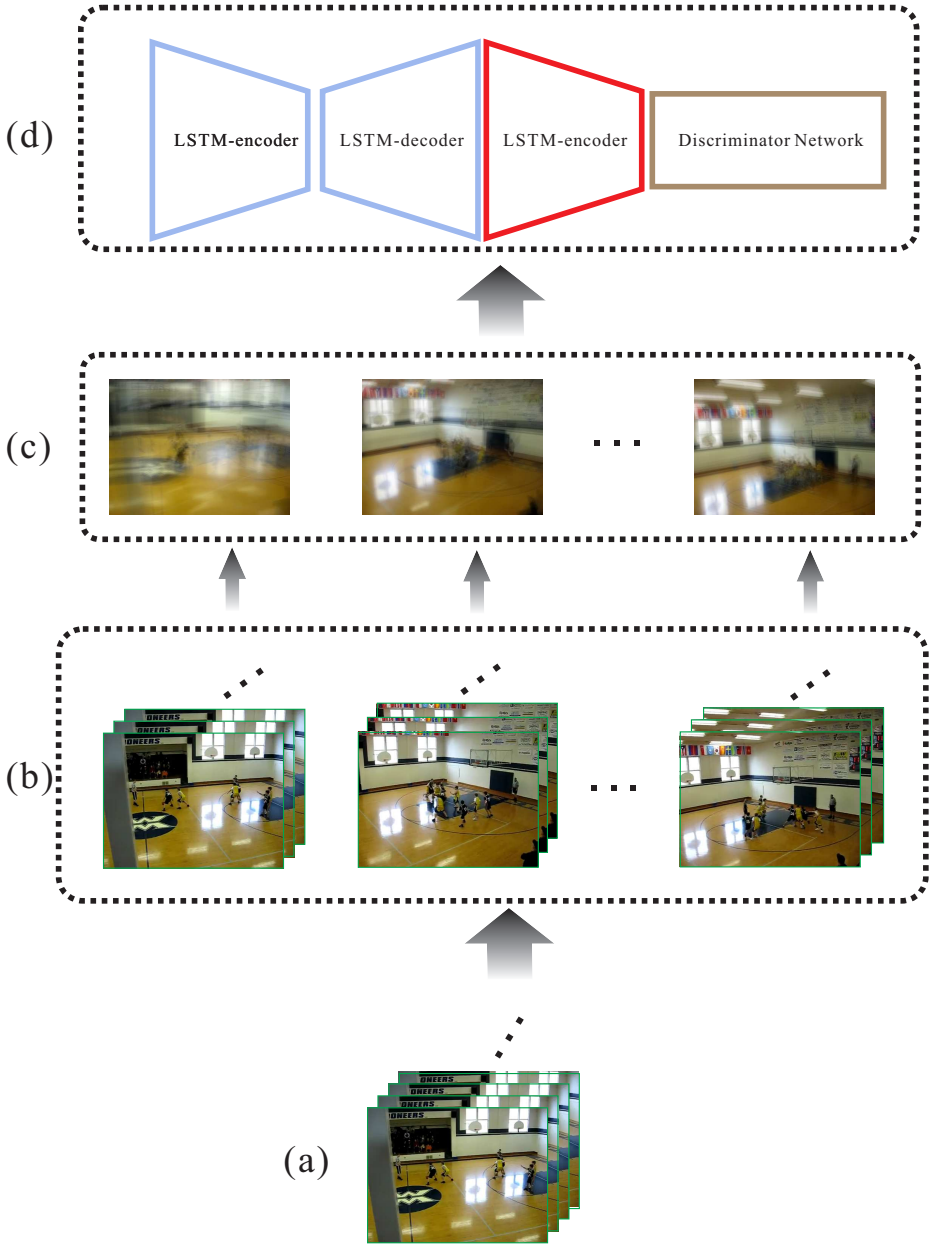
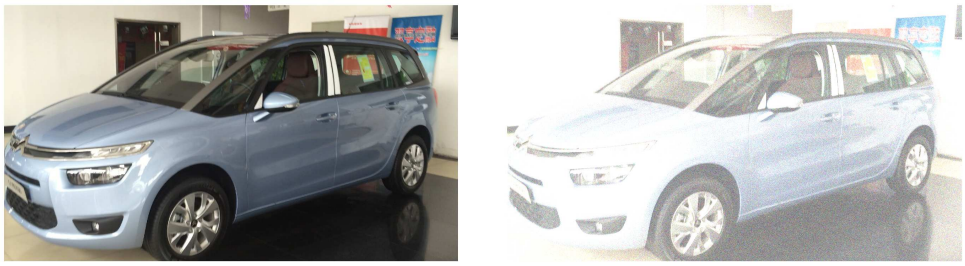


Figure 2.4: The workflow of the training. (a) Original video. (b) A fixed number of segments of the (a). (c) Extracting multiple dynamic images from (b). (d) LSTM-autoencoder-based adversarial network that is shown in Figure 2.6 in detail.

and allows the system to be more robust against noise. We propose to use two simple yet effective methods for video data augmentation: adding noise to the training samples and spatial augmentation of training samples.

Let X be the existing training samples and a Gaussian noise η is added to them. We obtain the increased training samples $(X + \eta)$. This data augmentation makes the system robust to noise and distortions in the training stage. Figure 2.5 shows the example of adding a Gaussian noise to the existing sample, after which the existing sample is corrupted.



(a) (b)

Figure 2.5: Adding noise to the training sample. (a) Original video frame. (b) The frame with noise.

Spatial augmentation of training samples reduces the effects of overfitting (Karpathy *et al.*, 2014). we first crop the center regions of all video frames and then resize them to the size of 252×256. After that, we randomly resample 192×192 region and finally randomly flip the frames horizontally with 50% probability. Based on these two methods for video data augmentation, there are enough video data for training.

To address the limitation that summarizing the whole video into a dynamic image may result in the temporal and spatial information loss, we propose to extract multiple dynamic images from a given video sequence. In detail, we segregate the given video

into snippets and then extract the dynamic images from every snippet. In this way, we also obtain a fixed number of dynamic images for each video, which benefits LSTM to process the sequence.

Training module. We propose LSTM-autoencoder-based adversarial learning model for anomalous video detection. Our idea is to design an end-to-end model for anomalous video detection in an adversarial scheme that has three subnetworks, i.e., the LSTM-autoencoder network, the discriminator network, and an additional LSTM-encoder network. Unlike the standard GANs where the generator network is to synthesize realistic data from the random input z to fool the discriminator, here we propose LSTM-autoencoder network (R) to replace the generator network, yielding a low-dimension latent representation of the normal samples and reconstructing the input sample for the discrimination task. The main advantage is to capture the normal data distribution effectively and then it is helpful for abnormal data detection. For a given abnormal sample, the reconstruction error would be high because the LSTM-autoencoder is only suitable for reconstructing the normal samples.

Another novel aspect is that we add an additional LSTM-encoder network to further enhance the adversariality because we minimize the distance between the two latent low-dimension latent representations obtained from the LSTM-autoencoder network and the additional LSTM-encoder network. This also allows the model to further capture the training data distribution. The difference between the two latent representations works as the abnormality detector. If the difference is high, the input sample is detected as an abnormality. Detailed descriptions are presented in the following subsections.

Anomalous video detection module. After training, the LSTM-autoencoder

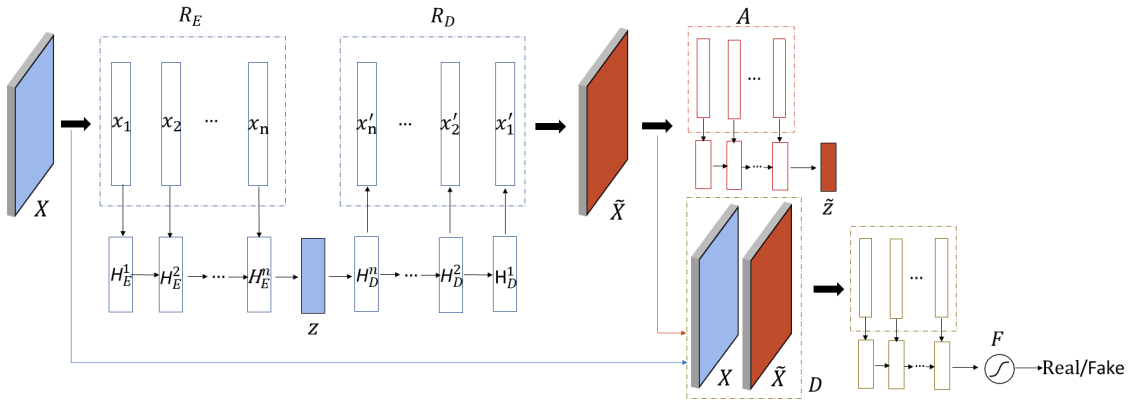


Figure 2.6: The architecture of the proposed LSTM-autoencoder-based adversarial network. It contains LSTM-autoencoder network, an additional LSTM-encoder network, and discriminator network. The input sequence (X) is obtained from multiple dynamic images. The encoder R_E based on LSTM reads the input sequence and learns the low-dimension latent representation z, after which the LSTM-decoder R_D reconstruct the input sequence. The discriminator network D distinguishes the original input sequence and the reconstructed sequence ($\tilde{X}$) as the real or the fake. The additional encoder network A has the same structure of R_E with different parameters, yielding the latent representation $\tilde{z}$ of $\tilde{X}$. Minimizing the distance between z and $\tilde{z}$ is beneficial to capture the distribution of normal samples effectively.

network is only suitable to reconstruct the normal samples because its parameters are obtained by training normal samples only. For a given abnormal sample as the input in the inference stage, the LSTM-autoencoder network fails to reconstruct it accurately and the reconstruction error would be very high. Hence, the input is very different from the reconstructed data. The additional LSTM-encoder network further encodes the reconstructed data into latent representation, so that the difference between the input and the reconstructed data is enlarged. Based on the reconstruction error, we detect the abnormal samples and achieve anomalous video detection.

2.7.2 Training for LSTM-autoencoder-based adversarial network

The workflow of the training. Figure 2.4 illustrates the workflow of the training process. In Figure 2.4, (a) is the normal samples in the format of video sequences. Considering that summarizing the whole video sequence into a single dynamic image may lead to the temporal and spatial information loss, we segregate the video into a fixed number of segments as shown in Figure 2.4 (b). Then, we extract multiple dynamic images from every segment and the multiple dynamic images are shown in Figure 2.4 (c), which extract the temporal evolution of the frames of the video. We feed the multiple dynamic images as the sequences into LSTM-autoencoder-based adversarial network in Figure 2.4 (d). Next, we present the detail of the proposed LSTM-autoencoder-based adversarial network.

The architecture of LSTM-autoencoder-based adversarial network. Figure 2.6 shows the architecture of LSTM-autoencoder-based adversarial network. The proposed LSTM-autoencoder-based adversarial learning model is composed of three

subnetworks: the LSTM-autoencoder network (R), the discriminator network (D), and an additional LSTM-encoder network (A). The input in the training stage is only normal samples.

LSTM-autoencoder network (R). First sub-network is the LSTM-autoencoder network for reconstructing the input sample, which includes the LSTM-encoder (R_E) and the LSTM-decoder (R_D). R_E takes the normal samples as the input. We denote an input sequence (multiple dynamic images from the same segment) as X = {x_1,x_2,...,x_n}, where n is the length of the sequence and $x_i (i = 1, 2, ..., n)$ is every dynamic image. For LSTM-encoder, n is also the time steps and H_E^i is the hidden state of R_E that is computed from the current input dynamic image and previous states. After encoding, the low-dimension latent representation of the input sequence is obtained and used to reconstruct the target sequence $\widetilde{X}$= $\{x'_1, x'_2, ..., x'_n\}$ that is the same as the input sequence but in reverse order (Srivastava *et al.*, 2015). The final state H_E^L of the LSTM encoder is the latent representation which is the initial state for LSTM-decoder. R_E and R_D are jointly trained to reconstruct the input sequence X as $\tilde{X}$ and yield the latent representation z. R encodes the input sequence into a low-dimension latent representation z and then yields the reconstructed sequence $\tilde{X}$. The LSTM-autoencoder network aims to minimize the L_1 distance between X and $\tilde{X}$ as follows:

$$L_R = \mathbb{E}_{X \sim p(X)} \| X - R(X) \|_1 \tag{2.7.1}$$

where $R(X)$ is the reconstructed sequence $\tilde{X}$.

Discriminator network (D). The second sub-network is the discriminator network D for distinguishing between the input sequence X and the constructed sequence

$\tilde{X}$. The structure is the same as the LSTM-encoder but the parameters are different. The output is connected to a softmax layer for discriminating the real or the fake (input sequence or reconstructed sequence). D allows the whole network to achieve adversarial scheme and the loss function is:

$$L_D = \mathbb{E}_{X \sim p(X)} \left\| F(X) - \mathbb{E}_{X \sim p(X)} F(R(X)) \right\|_2 \tag{2.7.2}$$

Additional LSTM-encoder network (A). The third sub-network is the additional LSTM-encoder network (A) that encodes the reconstructed sequence $\tilde{X}$ to a low-dimension latent representation $\tilde{z}$. It has the same process as the LSTM-encoder R_E but different parameters, and the dimension of $\tilde{z}$ is the same as that of z for comparison. The additional LSTM-encoder network is novel because it further enables R to reconstruct the input sequence and capture the distribution of the normal samples. The two latent representations are used in the objective function to minimize the encoder loss:

$$L_A = \mathbb{E}_{X \sim p(X)} \| R_E(X) - A(R(X)) \|_1 \tag{2.7.3}$$

where $A(R(X))$ is the latent representation ($\tilde{z}$) of reconstructed sequence and $R_E(X)$ is the latent representation (z) of the input sequence.

Equation (2.7.3) is also used in the inference stage. For a given abnormal sample as the input, the additional LSTM-encoder network further enlarges the reconstruction error. Because of the high difference (calculated by (2.7.3)) between z and $\tilde{z}$, the input sample is defined as a abnormal one.

The objective function for LSTM-autoencoder-based adversarial network. The objective function combines three loss functions corresponding to the three sub-networks (Akcay *et al.*, 2018):

$$L = w_R L_R + w_D L_D + w_A L_A \tag{2.7.4}$$

where w_R, w_D and w_A are the weighting parameters. L_R, L_D and L_A represent the three loss functions corresponding to the three subnetworks, respectively.

2.7.3 Anomalous video detection

In the training stage, our proposed adversarial model is trained on normal samples without abnormal samples and after training its parameters are only suitable for reconstructing normal samples. During the classification stage, both normal and abnormal samples are fed into the model. (1) For abnormal samples as the inputs, R poorly reconstructs them and after the additional LSTM-encoder (A) the reconstruction error is enlarged. Based on the high difference (calculated by (2.7.3)) between z and $\tilde{z}$, the input samples are defined as abnormal ones. (2) For normal samples as the inputs, the constructed samples are very similar to the inputs, so the reconstruction error is low. Based on the similarity between z and $\tilde{z}$, the input samples are defined as v ones.

The anomalous video detection (AVD) can be formulated by using R_E and A as:

$$AVD(X) = \begin{cases} \text{Normal Class} & \text{if } \Phi(z, \widetilde{z}) > \xi, \\ \\ \text{Abnormal Class} & \text{otherwise.} \end{cases} \tag{2.7.5}$$

where ξ is a pre-defined threshold and $\Phi(z, \widetilde{z}) = \|z - \widetilde{z}\|_1$

2.8 Experiments

In this section, our proposed system "VidAnomaly" is evaluated on two different datasets. The performance results are analyzed and are compared with other methods. We adopt the following well-known metrics: accuracy, precision, recall and F1 score for the system performance evaluation.

2.8.1 Dataset description

Figure 2.7: A few examples of the CCV dataset.

We evaluate VidAnomaly on Colunbia Consumer Video Database (CCV) (Jiang *et al.*, 2011) and e-Lab Video Dataset (e-VDS) (Culurciello and Canziani, 2017).

CCV dataset. The CCV dataset contains 9,317 YouTube videos over 20 semantic categories. The dataset was collected to be related to consumer's interest and originality of video content without post-editing. For the video contents, the dataset includes events (e.g. basketball, swimming, etc), objects (dog, cat, and bird) and scenes like beach and playground. There is a baseline system using features such as SIFT and STIP for evaluation, so the feature extraction from every video is provided. Each category has at least 200 videos. Some examples of the CCV dataset are shown in Figure 2.7. We randomly choose categories as the normal class and a certain number of abnormal samples are randomly selected from other categories.

Figure 2.8: Sample categories of the e-VDS dataset.

e-VDS dataset. The e-VDS dataset contains 2050 videos with 35 classes, which was collected from the real-world scenario with 10 seconds each. The dataset focuses on objects such as bicycle, car, sofa, etc. Every category has at least 37 videos and the raw data is directly provided. Sample categories of the e-VDS dataset are illustrated

in Figure 2.8. We also randomly choose categories as the normal class and randomly select other categories as the abnormal samples.

2.8.2 Implementation details

To balance the computational burden and accuracy, we segregate the original video into 9 segments and then extract multiple dynamic image from every segment. The LSTM-autoencoder-based adversarial network is trained by ADAM (Kingma and Ba, 2014) optimizer. Each LSTM has 1024 units. The learning rate of ADAM is set to 0.0002, and momentums $\beta_1 = 0.5$, $\beta_2 = 0.999$.

2.8.3 Competing methods

To evaluate the perfomance of VidAnomaly, we compare it with existing methods.

DSEBMs. Deep structured energy based models (DSEBMs) (Zhai *et al.*, 2016) is a state-of-the-art deep learning method based on the energy function for anomaly detection.

VAE. Variational Auto Encoder (VAE) (An and Cho, 2015) is a deep generative model for anomaly detection. An *et al.* proposed to utilize the generative characteristics of the variational autoencoder for deriving the reconstruction of the data to detect anomalies.

AnoGan. Schlegl *et al.* (Schlegl *et al.*, 2017) proposed to use deep generative adversarial networks for anomaly detection. By concurrently training the two models, i.e., a generative model and a discriminator, anomalies are identified on unseen data based on unsupervised training of a model on normal data.

EGBAD. Zenati *et al.* (Zenati *et al.*, 2018) leveraged GAN for anomaly detection,

and achieved state-of-the-art performance on different datasets.

2.8.4 Ablation study

Dynamic image. Dynamic image in VidAnomaly system summarizes the video segments into a visual representation in the format of 2D image and extracts the temporal evolution information of the frames in the video.

We compare VidAnomaly with and without dynamic image. For VidAnomaly without dynamic image, we select video frames with a constant time interval. As shown in Table 2.1, VidAnomaly without dynamic image is denoted as VidAnomaly (DI-) and VidAnomaly (DI-) fails to achieve good performance, becasue selecting video frames with a constant time interval resulting in the temporal and critical information loss.

Methods	Accuracy	Precision	Recall	F1
VidAnomaly (DI-)	0.8726	0.8312	0.8612	0.8459
VidAnomaly (LS-)	0.8406	0.8011	0.8212	0.8110
VidAnomaly	0.9416	0.9031	0.8845	0.8937

Table 2.1: Ablation Study of VidAnomaly on e-VDS dataset.

Therefore, we propose to extract dynamic image as an intermediate step to obtain temporal evolution information. Benefiting from dynamic image, VidAnomaly achieves better performance than that without dynamic image. This experiment validates the effectiveness and necessity of dynamic image for video representation, improving the performance of the system.

LSTM-autoencoder. LSTM-autoencoder further preserves the temporal information of the sequence and yields the low-dimension latent representation of the input

sequence. The representation is also used to reconstruct the input sequence and the reconstructed sequence is fed into the discriminator network to achieve adversarial learning. LSTM-autoencoder plays an important role in VidAnomaly.

To study the importance of LSTM-autoencoder in VidAnomaly, we compare VidAnomaly without (denoted as vidAnomaly (LS-)) and with LSTM-autoencoder as shown Table 2.1. We adopt CNNs to extract the feature of multiple dynamic images. Since the correlation of the input sequence is not used, VidAnomaly without LSTM-autoencoder degrades the performance of the system.

2.8.5 Comparison of different methods

Methods	Accuracy	Precision	Recall	F1
DSEBMs	0.8393	0.7953	0.8121	0.8036
VAE	0.8535	0.8125	0.8332	0.8227
AnoGan	0.8926	0.8756	0.8521	0.8636
EGBAD	0.9032	0.8842	0.8636	0.8737
Ours	0.9416	0.9031	0.8845	0.8937

Table 2.2: Performance evaluation on e-VDS dataset.

We show the comparison of different methods and the performance evaluation in Table 2.2 and Table 2.3. The experimental results clearly illustrate that VidAnomaly achieves superior performance over other methods. In particular, our proposed method achieves a high accuracy of around 0.94 on e-VDS dataset. Moreover, the accuracy and F1 score for CCV dataset are highest in these methods. CCV dataset is more complicated because of longer video clips and similar scenes in normal and abnormal samples. This indicates the potential use of VidAnomaly in practice. Although DSEBMs works well on multiple datasets, VidAnomaly still outperforms.

This is because we make full use of the latent representations both from the LSTM-autoencoder network and the additional LSTM-encoder to detect anomalies. VAE provides acceptable performance on some certain datasets. However, the main problem is that the fixed parametrization is usually oversimplified compared to the true complex distribution of real-world data. This is not a problem of VidAnomaly because the LSTM-autoencoder is forced to make the reconstructed samples to follow the distribution of true data and the model is more hierarchical. AnoGan and EGBAD, in most cases, reach good performance. However, its overall performance is still not as good as that of VidAnomaly, because the additional LSTM-encoder network further makes the reconstruction error to be enlarged. In this way, the difference between the two latent representations from LSTM-autoencoder network and the additional LSTM-encoder is very high for abnormal samples.

Methods	Accuracy	Precision	Recall	F1
DSEBMs	0.8026	0.7432	0.7018	0.7219
VAE	0.7812	0.7022	0.642	0.6707
AnoGan	0.8211	0.7132	0.7563	0.7341
EGBAD	0.8132	0.7412	0.7023	0.7212
Ours	0.8303	0.7508	0.7326	0.7416

Table 2.3: Performance evaluation on CCV dataset.

2.8.6 Further analysis

In this subsection, we further demonstrate the reconstruction results for abnormal samples and the distribution of the anomaly scores during the inference phase. This allows us to better understand how VidAnomaly separates the abnormal samples from the normal ones.

Figure 2.9 illustrates how the LSTM-autoencoder reconstructs the abnormal samples. For any given abnormal sample that does not follow the distribution of target class, R is confused and reconstructs the abnormal sample with an unseen distribution. In this case, the LSTM-autoencoder fails to reconstruct the given abnormal sample accurately because the model is trained on normal samples and its parameters are only suitable for reconstructing the normal samples. In Figure 2.9, the normality is the car and the first row is abnormal samples (watch, barcode, and chair). The second row is the results of reconstruction and we can see that the LSTM-autoencoder poorly reconstructs them and the reconstructed results are still similar to the normal samples. Based on the high reconstruction error, the abnormal samples are detected.

Figure 2.10 provides the distribution of the anomaly scores during the inference stage. We can observe how the normal samples are distinguished from the abnormal ones. The range of the anomaly score is from 0 to 1 and the score obtained by the difference between the two latent representations from R_E and A, where 0 represents the reconstructed sequence is the same as the input sequence and the input sample is absolutely normal, while 1 represents the input sample is absolutely abnormal. The hard-to-decide cases lie in the reject region, which requires human intervention. Figure 2.10 (a) shows the anomaly scores on the e-VDS dataset and Figure 2.10 (b) provides the anomaly score on the CCV dataset. It can be seen that the reject region is larger on CCV dataset, which means that the CCV dataset is more complicated and there are more normal samples and abnormal samples that are difficult to distinguish in this dataset. Although there are small overlaps, the normals and abnormals are distinguished in most of data with good performance. This further shows the effectiveness of VidAnormaly on anomalous video detection.

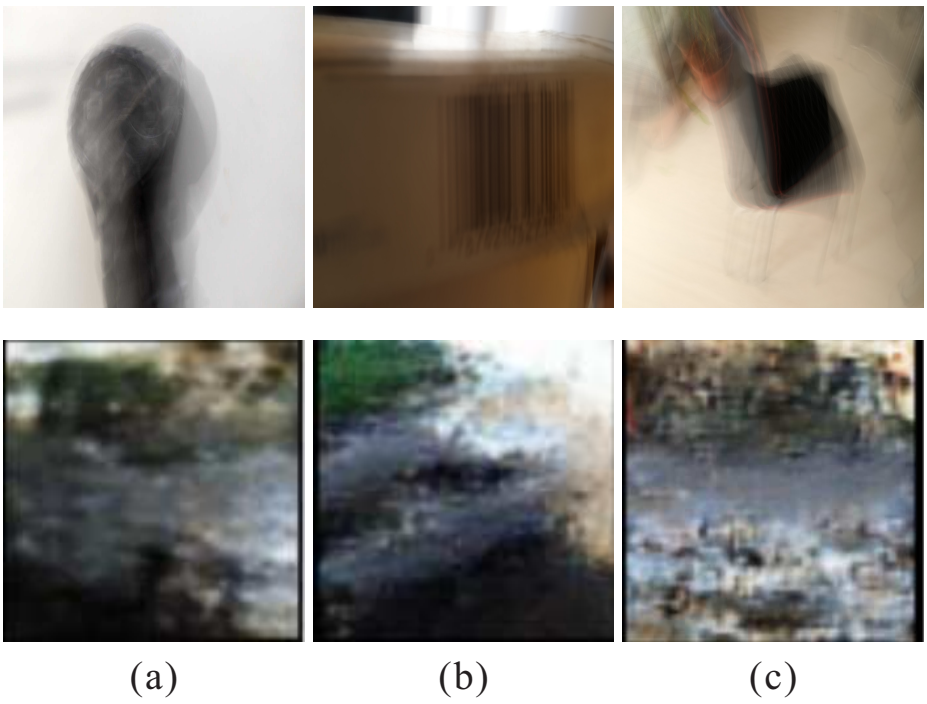


Figure 2.9: The results of reconstruction for abnormal samples. The first row is the example of input dynamic image. The second row shows the results of reconstruction. The normality is car. Based on the high reconstruction error, the abnormal samples are detected.

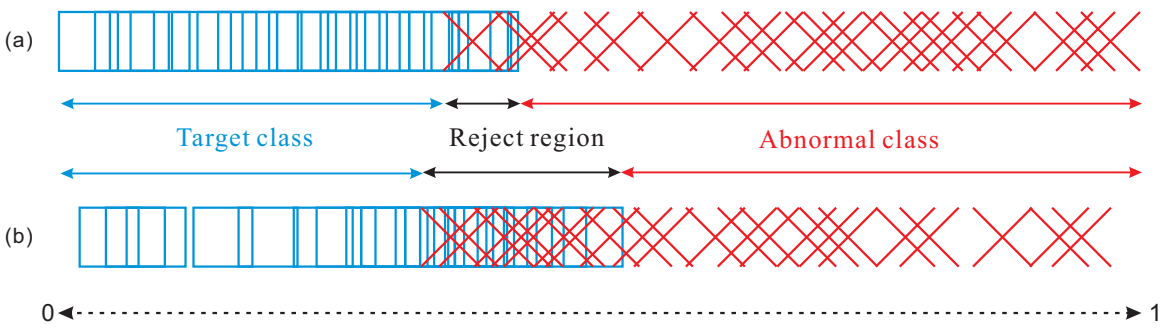


Figure 2.10: Distribution of the scores for both normal and abnormal test samples. The blue squares are normal samples, while the red "×" represents the abnormal samples. (a) The scores for e-VDS dataset. (b) The scores for CCV dataset.

2.9 Conclusion

In this paper, we propose and develop a LSTM-autoencoder-based adversarial learning system, called VidAnomaly. Rather than directly processing every video frame or extracting feature vectors from video, we introduce dynamic image to summarize the video contents into a 2D image that encodes the temporal evolution and preserve spatial information of the video content. To keep the temporal information as much as possible, we segregate the video into segments and extract multiple dynamic images from them. Without the abnormal samples in the training stage, we propose LSTM-autoencoder-based adversarial learning network. The input in our task is the sequence and LSTM-autoencoder yields the low-dimension latent representation and reconstruct the input sequence for the discriminator network to achieve adversarial learning. The novelty of LSTM-autoencoder-based adversarial learning network is to add an additional LSTM-encoder network that encoders the reconstructed sequence to obtain the latent representation. The difference between the two latent representation is further enlarged when the input is abnormal samples. LSTM-autoencoder-based adversarial learning network is trained on normal samples only and its parameter is only suitable for reconstructing normal samples. In the inference stage, for a given abnormal sample, the model poorly reconstructs the input sample and the reconstruction error would be high. Based on the high reconstruction error, we achieve anomalous video detection without abnormalities in the training stage. Experiments and analysis corroborate that different parts of VidAnomaly play a vital role in anomalous video detection.

Bibliography

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., *et al.* (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

Abe, N., Zadrozny, B., and Langford, J. (2006). Outlier detection by active learning. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 504–509.

Adam, A., Rivlin, E., Shimshoni, I., and Reinitz, D. (2008). Robust real-time unusual event detection using multiple fixed-location monitors. *IEEE transactions on pattern analysis and machine intelligence*, **30**(3), 555–560.

Akcay, S., Atapour-Abarghouei, A., and Breckon, T. P. (2018). Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian conference on computer vision*, pages 622–637. Springer.

An, J. and Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, **2**(1), 1–18.

Ashfaq, R. A. R., Wang, X.-Z., Huang, J. Z., Abbas, H., and He, Y.-L. (2017).

Fuzziness based semi-supervised learning approach for intrusion detection system. *Information Sciences*, **378**, 484–497.

Barnett, T., Jain, S., Andra, U., and Khurana, T. (2018). Cisco visual networking index (vni) complete forecast update, 2017–2022. *Americas/EMEAR Cisco Knowledge Network (CKN) Presentation*.

Barrett, D. (2013). One surveillance camera for every 11 people in britain, says cctv survey.

Beal, J., Kim, E., Tzeng, E., Park, D. H., Zhai, A., and Kislyuk, D. (2020). Toward transformer-based object detection. *arXiv preprint arXiv:2012.09958*.

Bertasius, G., Wang, H., and Torresani, L. (2021). Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*.

Bilen, H., Fernando, B., Gavves, E., Vedaldi, A., and Gould, S. (2016). Dynamic image networks for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3034–3042.

Blair, C. G. and Robertson, N. M. (2015). Video anomaly detection in real time on a power-aware heterogeneous platform. *IEEE Transactions on Circuits and Systems for Video Technology*, **26**(11), 2109–2122.

Cai, S., Zuo, W., Davis, L. S., and Zhang, L. (2018). Weakly-supervised video summarization using variational encoder-decoder and web prior. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 184–200.

Camacho, J., Pérez-Villegas, A., García-Teodoro, P., and Maciá-Fernández, G.

(2016). Pca-based multivariate statistical network monitoring for anomaly detection. *Computers & Security*, **59**, 118–137.

Campos, G. O., Zimek, A., Sander, J., Campello, R. J., Micenková, B., Schubert, E., Assent, I., and Houle, M. E. (2016). On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. *Data mining and knowledge discovery*, **30**(4), 891–927.

Canel, C., Kim, T., Zhou, G., Li, C., Lim, H., Andersen, D. G., Kaminsky, M., and Dulloor, S. R. (2019). Scaling video analytics on constrained edge nodes. *arXiv preprint arXiv:1905.13536*.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer.

Chang, J., Wang, L., Meng, G., Xiang, S., and Pan, C. (2017). Deep adaptive image clustering. In *Proceedings of the IEEE international conference on computer vision*, pages 5879–5887.

Chen, C., Liu, J., Xie, Y., Ban, Y. X., Wu, C., Tao, Y., and Song, H. (2020). Latent regularized generative dual adversarial network for abnormal detection. In *IJCAI*, pages 760–766.

Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., and Gao, W. (2021). Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12299–12310.

Chen, T. Y.-H., Ravindranath, L., Deng, S., Bahl, P., and Balakrishnan, H. (2015). Glimpse: Continuous, real-time object recognition on mobile devices. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 155–168. ACM.

Chen, Y., He, W., Hua, Y., and Wang, W. (2016). Compoundeyes: Near-duplicate detection in large scale online video systems in the cloud. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE.

Chong, Y. S. and Tay, Y. H. (2017). Abnormal event detection in videos using spatiotemporal autoencoder. In *International Symposium on Neural Networks*, pages 189–196. Springer.

Cordonnier, J.-B., Loukas, A., and Jaggi, M. (2019). On the relationship between self-attention and convolutional layers. In *International Conference on Learning Representations*.

Culurciello, E. and Canziani, A. (2017). e-Lab video data set. `https://engineering.purdue.edu/elab/eVDS/`.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Donahue, J., Krähenbühl, P., and Darrell, T. (2016). Adversarial feature learning. *arXiv preprint arXiv:1605.09782*.

Doshi, K. and Yilmaz, Y. (2020). Fast unsupervised anomaly detection in traffic videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 624–625.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., *et al.* (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Fernando, B., Gavves, E., Oramas, J. M., Ghodrati, A., and Tuytelaars, T. (2015). Modeling video evolution for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5378–5387.

Fernando, B., Anderson, P., Hutter, M., and Gould, S. (2016a). Discriminative hierarchical rank pooling for activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1924–1932.

Fernando, B., Gavves, E., Oramas, J., Ghodrati, A., and Tuytelaars, T. (2016b). Rank pooling for action recognition. *IEEE transactions on pattern analysis and machine intelligence*, **39**(4), 773–787.

Gaddam, S. R., Phoha, V. V., and Balagani, K. S. (2007). K-means+ id3: A novel method for supervised anomaly detection by cascading k-means clustering and id3 decision tree learning methods. *IEEE transactions on knowledge and data engineering*, **19**(3), 345–354.

Gardner, A. B., Krieger, A. M., Vachtsevanos, G., Litt, B., and Kaelbing, L. P. (2006). One-class novelty detection for seizure analysis from intracranial eeg. *Journal of Machine Learning Research*, **7**(6).

Gavrilyuk, K., Sanford, R., Javan, M., and Snoek, C. G. (2020). Actor-transformers for group activity recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 839–848.

Ghasedi, K., Wang, X., Deng, C., and Huang, H. (2019). Balanced self-paced learning for generative adversarial clustering network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4391–4400.

Girdhar, R., Carreira, J., Doersch, C., and Zisserman, A. (2019). Video action transformer network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–253.

Gkioxari, G. and Malik, J. (2015). Finding action tubes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 759–768.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, **27**.

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y., *et al.* (2022). A survey on vision transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Hasan, M., Choi, J., Neumann, J., Roy-Chowdhury, A. K., and Davis, L. S. (2016).

Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 733–742.

Hinami, R., Mei, T., and Satoh, S. (2017). Joint detection and recounting of abnormal events by learning deep generic knowledge. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3619–3627.

Hoai, M. and Zisserman, A. (2014). Improving human action recognition using score distribution and ranking. In *Asian conference on computer vision*, pages 3–20. Springer.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Ionescu, R. T., Smeureanu, S., Popescu, M., and Alexe, B. (2019). Detecting abnormal events in video using narrowed normality clusters. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1951–1960. IEEE.

Ji, S., Xu, W., Yang, M., and Yu, K. (2012). 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, **35**(1), 221–231.

Jiang, Y., Chang, S., and Wang, Z. (2021). Transgan: Two pure transformers can make one strong gan, and that can scale up. *Advances in Neural Information Processing Systems*, **34**.

Jiang, Y.-G., Ye, G., Chang, S.-F., Ellis, D., and Loui, A. C. (2011). Consumer video understanding: A benchmark database and an evaluation of human and

machine performance. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, pages 1–8.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., and Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732.

Khan, S. S. and Madden, M. G. (2014). One-class classification: taxonomy of study and review of techniques. *The Knowledge Engineering Review*, **29**(3), 345–374.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Welling, M. (2014). Stochastic gradient vb and the variational auto-encoder. In *Second International Conference on Learning Representations, ICLR*, volume 19, page 121.

Kriegel, H.-P., Schubert, M., and Zimek, A. (2008). Angle-based outlier detection in high-dimensional data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 444–452.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, **25**, 1097–1105.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, **1**(4), 541–551.

Lee, J., Reade, W., Sukthankar, R., Toderici, G., *et al.* (2018). The 2nd youtube-8m large-scale video understanding challenge. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0.

Lee, K., Chang, H., Jiang, L., Zhang, H., Tu, Z., and Liu, C. (2021). Vitgan: Training gans with vision transformers. *arXiv preprint arXiv:2107.04589*.

Li, S. and He, W. (2019). Vidanomaly: Lstm-autoencoder-based adversarial learning for one-class video classification with multiple dynamic images. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 2881–2890. IEEE.

Liu, W., Luo, W., Lian, D., and Gao, S. (2018). Future frame prediction for anomaly detection–a new baseline. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6536–6545.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, **60**(2), 91–110.

Mahasseni, B., Lam, M., and Todorovic, S. (2017). Unsupervised video summarization with adversarial lstm networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 202–211.

Ng, A. *et al.* (2011). Sparse autoencoder. *CS294A Lecture notes*, **72**(2011), 1–19.

Pakha, C., Chowdhery, A., and Jiang, J. (2018). Reinventing video streaming for distributed vision analytics. In *10th {USENIX} Workshop on Hot Topics in Cloud Computing (HotCloud 18)*.

Ren, M., Zeng, W., Yang, B., and Urtasun, R. (2018). Learning to reweight examples

for robust deep learning. In *International Conference on Machine Learning*, pages 4334–4343. PMLR.

Ryoo, M. S., Rothrock, B., and Matthies, L. (2015). Pooled motion features for first-person videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 896–904.

Sabokrou, M., Fathy, M., and Hoseini, M. (2016). Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder. *Electronics Letters*, **52**(13), 1122–1124.

Sabokrou, M., Khalooei, M., Fathy, M., and Adeli, E. (2018). Adversarially learned one-class classifier for novelty detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3379–3388.

Sakurada, M. and Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, pages 4–11.

Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., and Langs, G. (2017). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, pages 146–157. Springer.

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, **404**, 132306.

Siffer, A., Fouque, P.-A., Termier, A., and Largouet, C. (2017). Anomaly detection in streams with extreme value theory. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1067–1075.

Simonyan, K. and Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*.

Smola, A. J. and Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, **14**(3), 199–222.

Srivastava, N., Mansimov, E., and Salakhudinov, R. (2015). Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852.

Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. (2015). Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wang, J., Feng, Z., Chen, Z., George, S., Bala, M., Pillai, P., Yang, S.-W., and Satyanarayanan, M. (2018a). Bandwidth-efficient live video analytics for drones via edge computing. In *2018 IEEE/ACM Symposium on Edge Computing (SEC)*, pages 159–173. IEEE.

Wang, J., Zhou, W., Tang, J., Fu, Z., Tian, Q., and Li, H. (2018b). Unregularized auto-encoder with generative adversarial networks for image generation. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 709–717.

Wang, X., Girshick, R., Gupta, A., and He, K. (2018c). Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803.

Weissenborn, D., Täckström, O., and Uszkoreit, J. (2019). Scaling autoregressive video models. *arXiv preprint arXiv:1906.02634*.

Xiao, Y., Jia, Y., Liu, C., Cheng, X., Yu, J., and Lv, W. (2019). Edge computing security: State of the art and challenges. *Proceedings of the IEEE*, **107**(8), 1608–1631.

Xu, D., Ricci, E., Yan, Y., Song, J., and Sebe, N. (2015). Learning deep representations of appearance and motion for anomalous event detection. *arXiv preprint arXiv:1510.01553*.

Yang, H., Wang, B., Lin, S., Wipf, D., Guo, M., and Guo, B. (2015). Unsupervised extraction of video highlights via robust recurrent auto-encoders. In *Proceedings of the IEEE international conference on computer vision*, pages 4633–4641.

Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., and Toderici, G. (2015). Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702.

Zenati, H., Foo, C. S., Lecouat, B., Manek, G., and Chandrasekhar, V. R. (2018). Efficient gan-based anomaly detection. *arXiv preprint arXiv:1802.06222*.

Zhai, S., Cheng, Y., Lu, W., and Zhang, Z. (2016). Deep structured energy based models for anomaly detection. In *International Conference on Machine Learning*, pages 1100–1109. PMLR.

Zhang, T., Chowdhery, A., Bahl, P. V., Jamieson, K., and Banerjee, S. (2015). The design and implementation of a wireless video surveillance system. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pages 426–438. ACM.

Zhang, Y., Liang, X., Zhang, D., Tan, M., and Xing, E. P. (2018). Unsupervised object-level video summarization with online motion auto-encoder. *Pattern Recognition Letters*.

Zhao, B., Fei-Fei, L., and Xing, E. P. (2011). Online detection of unusual events in videos via dynamic sparse coding. In *CVPR 2011*, pages 3313–3320. IEEE.

Zhao, H., Jia, J., and Koltun, V. (2020). Exploring self-attention for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10076–10085.

Zhao, Y., Deng, B., Shen, C., Liu, Y., Lu, H., and Hua, X.-S. (2017). Spatio-temporal autoencoder for video anomaly detection. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1933–1941. ACM.

Chapter 3

OED: Onsite Event Detection from Surveillance Videos Based on Vision Transformer in Adversarial Learning

3.1 Abstract

With the increasing deployment of surveillance cameras, large volumes of video data are continuously recorded. This puts great pressure on network and storage resources, and it is infeasible to transmit all video data from cameras to the cloud servers. Instead, we prefer to filter out irrelevant contents and only transmit the events of interests on the edge computing devices, so as to suppress the network traffic and reduce storage space. The existing approaches usually rely on pre-trained models

to detect events. However, it is hard to enumerate all events of interest beforehand. Obtaining suitable training data is also challenging. Therefore, the pre-trained model is not reliable or practical. In this paper, we propose Onsite Event Detection (OED), a system that enables real-time event detection on edge. OED first trains a transformer-based autoencoder in adversarial learning to learn the spatial-temporal representation of video data observed recently. Then it gains the ability to differentiate eccentric data patterns of events from routine. OED also features an updating strategy that adapts to the changing environment dynamically. As such, OED is capable of continuously detecting events in video streams. Experimental results show that OED effectively and efficiently detects events and substantially saves bandwidth.

3.2 Introduction

With the development of surveillance systems, millions of video cameras have been deployed (Barrett, 2013) for security and safety purposes. These always-on cameras continuously record large volumes of video data. Surveillance videos are uploaded to cloud servers to be processed and analyzed. However, large segments of videos only contain background or irrelevant contents such as empty streets, static scenes, swaying tree leaves or moving but noisy routines. These video contents are meaningless but place severe stress on network resources. People are usually interested in video contents containing events of interest which are defined as activities or behaviors that deviate from the routine. These events are also environmentally dependent (e.g., a car appearing in an empty street is an event, while the regular traffic flow on the highway is not). Our key observation is that surveillance video is highly redundant, containing a large number of frames with temporal similarity. It brings new opportunities

to significantly reduce bandwidth usage.

In recent years, "Edge Computing" has attracted much attention from researchers, and it pushes the computation and storage resources closer to the location where it is needed. In addition, edge computing also improves response times and saves bandwidth (Xiao *et al.*, 2019). With the advancement of IoT and its application in smart cities, a huge number of terminal devices are deployed around the streets, highways and factories for security surveillance. As illustrated in Figure 3.1, the various edge devices are deployed and concurrently controlled by the cloud server. In the existing and advanced surveillance system, the distributed monitors obtain images or videos, sending a large volume of surveillance videos to cloud servers or data centers, which brings problems including increasing the bandwidth consumption, backbone network congestion, affecting real-time analysis, etc.

We anticipate that Onsite Event Detection from surveillance videos based on vision transformer in adversarial learning will be widely applicable in the scenarios like monitoring in smart homes, smart cities and Internet of Things due to real-time analysis and in-situ detection capacity.

Despite its practical values, event detection on edge encounters several critical challenges:

- *Limited edge computing resources.* Edge computing devices usually have limited processing power compared to resource-rich data centers. Since running the vision pipeline is typically computationally expensive, how to efficiently detect events on edge is challenging.

- *Requirements of online reliable detection.* There are diverse events captured by cameras, reflecting what is happening in daily life. Servers are expected to get

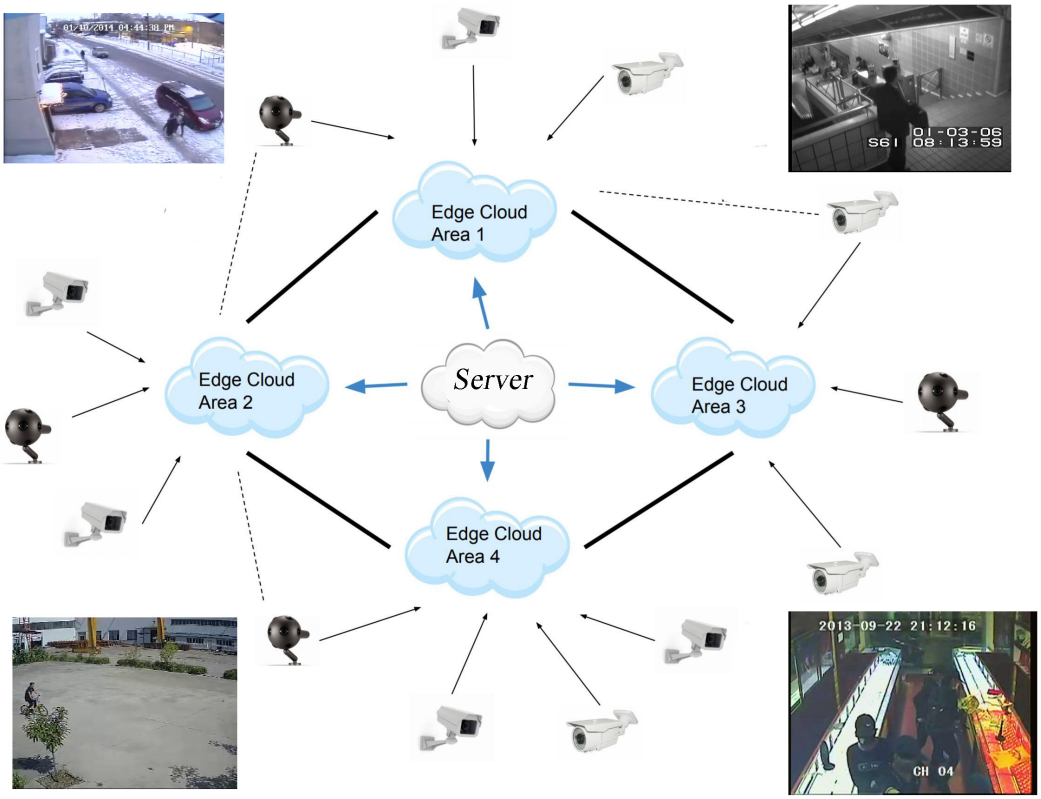


Figure 3.1: Real-time surveillance video analysis using edge computing.

access to all the useful information for generating different real-time insights (e.g., traffic monitoring, people tracking, activity inference). It is nontrivial to identify events accurately and in an online manner.

- *Changing environment in surveillance.* In real-world scenarios, a single camera may capture the changing environment. For example, the background is different after rain or snow; the traffic flow on the highway increases a lot at rush hour; the color of tree leaves changes gradually from spring to autumn. The detector should dynamically adapt to the changing environment in video streams.

- *Ambiguous definition of anomaly events.* For example, the use of bicycles and vehicles could be illegal in a pedestrian-only area but are legal and normal in other scenarios. The detection of anomaly events is on a case-by-case basis.

- *Dependence on highly-curated datasets.* There are not sufficient annotated abnormal samples due to the rare occurrences of anomalies in real-life situations. Besides, anomalous events are unbounded in real-world situations, so it is almost impossible to collect all kinds of anomalies.

In this paper, we aim to achieve efficient video event detection and filter out irrelevant contents on edge devices. In our system, the large amount of video data do not have to be uploaded to the cloud servers or data centers. By uploading only necessary data to servers, we significantly reduce the bandwidth consumption without sacrificing the surveillance performance, hence it incurs low backbone bandwidth consumption and latency.

The existing approaches (Canel *et al.*, 2019; Wang *et al.*, 2018a) rely on pre-trained models to detect specific events in a supervised way. Since it is challenging to enumerate all events of interest, the pre-trained models may leave out undefined but relevant events. Recently, many studies have investigated abnormal events detection that is based on an autoencoder in videos (Zhao *et al.*, 2017; Hasan *et al.*, 2016; Zhao *et al.*, 2011), which achieves good performance.

Our architecture is based on an autoencoder in unsupervised learning. It learns the normal pattern via an autoencoder and then reconstructs normal events with a small reconstruction error before detection in the training stage. We obtain the trained model in the training stage and abnormal events are detected based on a large reconstruction error in the inference stage. Therefore, OED with an autoencoder is able to perform online reliable detection and rely less on the datasets. This is because only normal samples are required to learn a model to represent normal events in the training stage. Thus, there is no need to collect abnormal samples for training. Also, the normal and abnormal events are defined without ambiguousness since we only collect enough normal events for training and the events that do not belong to normal events are abnormal.

We notice that the autoencoder with convolutional neural networks (CNNs) (LeCun *et al.*, 1989) is currently dominating in anomaly detection in computer vision tasks (Li and He, 2019; Sabokrou *et al.*, 2018; Ionescu *et al.*, 2019). However, there are a few inherent limitations of convolution operators in video tasks (Bertasius *et al.*, 2021). 1) CNNs have strong inductive biases like local connectivity and translation equivariance. 2) Kernels of convolution are specifically designed to obtain short-range spatial and temporal information, so it is difficult to capture long-range dependencies

that extend beyond the receptive field among video frames.

To address the issues above, we propose Vision Transformer autoencoder to process videos for our task. Vision Transformer (ViT) has shown competitive performance on computer vision tasks such as object detection (Beal *et al.*, 2020), action recognition in video (Bertasius *et al.*, 2021), multitask pretraining Chen *et al.* (2021). ViT without using convolution or pooling is based on Transformer (Vaswani *et al.*, 2017) and it outperforms CNNs if the dataset for pretraining is large enough (Dosovitskiy *et al.*, 2020). ViT (Dosovitskiy *et al.*, 2020) interprets an image as a sequence of tokens (analogous to words in natural language processing (NLP)), which achieves comparable classification accuracy with smaller computational budgets on the ImageNet (Deng *et al.*, 2009). ViT relies on globally-contextualized representation with self-attention pattern and demonstrating advantages in non-local contextual dependencies (Lee *et al.*, 2021) and efficiency. Recent work in the still image domain (Carion *et al.*, 2020; Dosovitskiy *et al.*, 2020; Zhao *et al.*, 2020) has shown that ViTs are faster at training and inference than CNNs, enabling better learning capacities for the same computational budget. In our task, the aim is to establish an efficient and real-time system for abnormal video event detection and saving bandwidth on edge. Therefore, the architecture based on ViT autoencoder is more applicable to our system.

Another problem of autoencoder-based methods is that it is difficult to guarantee that larger reconstruction errors for abnormal events necessarily happen using trained model Liu *et al.* (2018) because as the complexity of the images increase, it is difficult for autoencoders to obtain high-quality reconstruction and reconstructed images start to get blurry (Wang *et al.*, 2018b). Therefore, we propose to integrate generative adversarial networks (GAN) into our system to establish GAN-style autoencoder

architecture for enlarging the reconstruction errors for abnormal samples. The adversarial learning mechanism is beneficial for capturing the normal data distribution as much as possible in the training stage by distinguishing between input data and reconstructed one. After adversarial training, the ViT-based autoencoder reconstructs the normal samples with a minimum reconstruction error to successfully fool the discriminator. Based on this, the ViT-based autoencoder only learns the distribution of normal samples and the reconstruction error corresponding to normal events is small. In the inference stage, reconstruction errors corresponding to abnormal events would be large enough to be detected. Hence, we achieve an unsupervised learning manner by adversarial training (only normal samples are required for training).

Besides, OED continuously updates the model using new observations, allowing it to adapt to new event patterns dynamically in the environment. Updating has little time cost because re-training the model with a small number of training samples only takes fewer iterations to converge. As such, OED executes on edge with only CPU power. It detects up to 90% events and reduces bandwidth consumption by 2 to 15 times depending on surveillance scenarios and parameter selection.

We summarize the following components to address the challenges of our task: (1) We propose and build a real-time surveillance system based on ViT autoencoder in adversarial learning, which improves response times and saves bandwidth. (2) The system architecture coherently integrates ViT-based autoencoder to model the temporal correlation among frames based on an unsupervised manner, defining the normal and abnormal samples without ambiguousness, which also performs online reliable detection and relies less on the highly-curated datasets. (3) Our system "OED" continuously updates the model using new observations with little time cost

for adapting to the dynamical environment in various surveillance scenarios.

The rest of this chapter is organized as follows. A brief review on related work is given in Section 3.3. Section 3.4 presents the design and model for OED system. Section 3.5 explains the implementation details and demonstrates the experimental results. Finally, the conclusion follows in Section 3.6.

3.3 Related Work

Researchers have explored efficient use of bandwidth on camera networks (Pakha *et al.*, 2018; Zhang *et al.*, 2015; Chen *et al.*, 2015). Zhang et al. (Zhang *et al.*, 2015) designed a wireless distributed surveillance system that minimizes bandwidth usage through object-based frame selection and content-aware traffic schedule. However, this system was designed to suppress redundancy among cameras in a cluster. Redundant data from individual cameras are not compressed. Pakha et al. (Pakha *et al.*, 2018) proposed protocols to optimize bandwidth using superposition coding. The protocols entrust servers to decide what/when to upload from cameras. Although it saves bandwidth significantly, the delay cannot be estimated because of the iterative communication workflow and server-side latency. In contrast, our work aims to save bandwidth by detecting events of interest and transmitting detected events only.

Efforts have been made to filter relevant events on edge. Pre-trained models are generally used for detecting specific events. Canel et al. (Canel *et al.*, 2019) designed an edge-to-cloud system using edge filters to upload only relevant video frames. They utilized "micro-classifiers" to detect events on edge nodes. Wang et al. (Wang *et al.*, 2018a) explored an early discard strategy to select frames to send between the drone and a cloudlet. This system performs detection using DNNs and classifiers (SVMs) for

each task. Our work shares their strategy of discarding irrelevant frames for saving bandwidth. However, since it is not possible to enumerate all events beforehand, pre-trained model are not suitable to detect generic events of interest.

Previous works on video anomaly detection are also relevant to our work. Zhao et al. (Zhao *et al.*, 2011) proposed a sparse coding approach with the hand-crafted feature to detect unusual events in videos. However, their work is built on low-level features, and are not in a setting of edge computing. Hasan et al. (Hasan *et al.*, 2016) leveraged a fully convolutional autoencoder to learn regularity across the video data. Further, 3D convolutional autoencoder (Zhao *et al.*, 2017) and convolutional LSTM autoencoder (Chong and Tay, 2017) have been used to learn spatio-temporal normal patterns. Hinami et al. (Hinami *et al.*, 2017) explored joint detection of abnormal events in videos by plugging the CNN model into anomaly detectors. These approaches have not explored online detection, requiring the video data to train the model beforehand.

Recently, autoencoders have also been applied to other video processing tasks such as video summarization (Zhang *et al.*, 2018; Cai *et al.*, 2018; Mahasseni *et al.*, 2017), video highlight detection (Yang *et al.*, 2015), and video representation (Srivastava *et al.*, 2015). For instance, Zhang et al. (Zhang *et al.*, 2018) proposed to generate short video summaries by employing computationally expensive object detection and tracking to obtain super-segmented "motion clips", and then processing with an online motion autoencoder. With different application scenarios, we here explore the efficient architecture to detect events of interest and filter out the irrelevant events.

Our method is more closely related to leveraging self-attention as a substitute for convolution (Ren *et al.*, 2018; Cordonnier *et al.*, 2019; Zhao *et al.*, 2020). In these

work, they use individual pixels as queries. To keep a low memory and an acceptable computational cost, they must restrict the scope of self-attention to locality or employ global self-attention on heavily downsized versions of the image. An alternative strategy is Vision Transformers (Dosovitskiy *et al.*, 2020) which were shown to produce outstanding results image classification. ViT has been recently adopted for video generation (Weissenborn *et al.*, 2019). We note that ViT has been used on top of convolutional feature maps for action localization and recognition (Girdhar *et al.*, 2019), video classification (Wang *et al.*, 2018c), and group activity recognition (Gavrilyuk *et al.*, 2020).

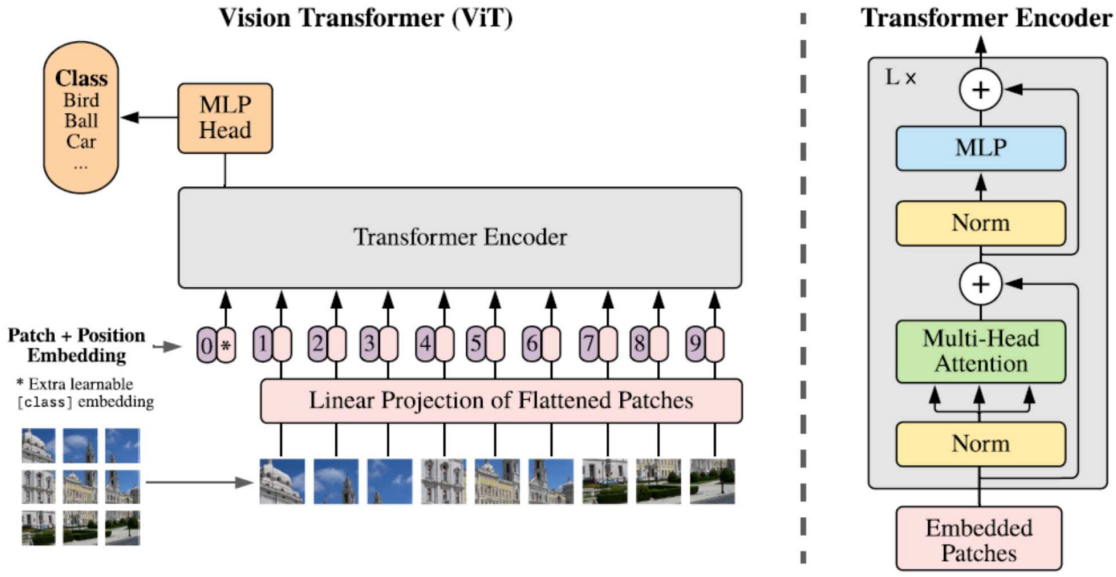


Figure 3.2: The structure of ViT (Dosovitskiy *et al.*, 2020).

Vision Transformer (ViT) (Dosovitskiy *et al.*, 2020) is a pure transformer directly applicable to image classification task using the sequences of image patches. Figure 3.2 shows the structure of ViT. To handle 2D images, the image is projected into a

sequence of flattened 2D patches. Because the transformer has fixed widths in all of its layers, a trainable linear projection maps the vectorized patch to the given dimension, the output of which is the patch embeddings.

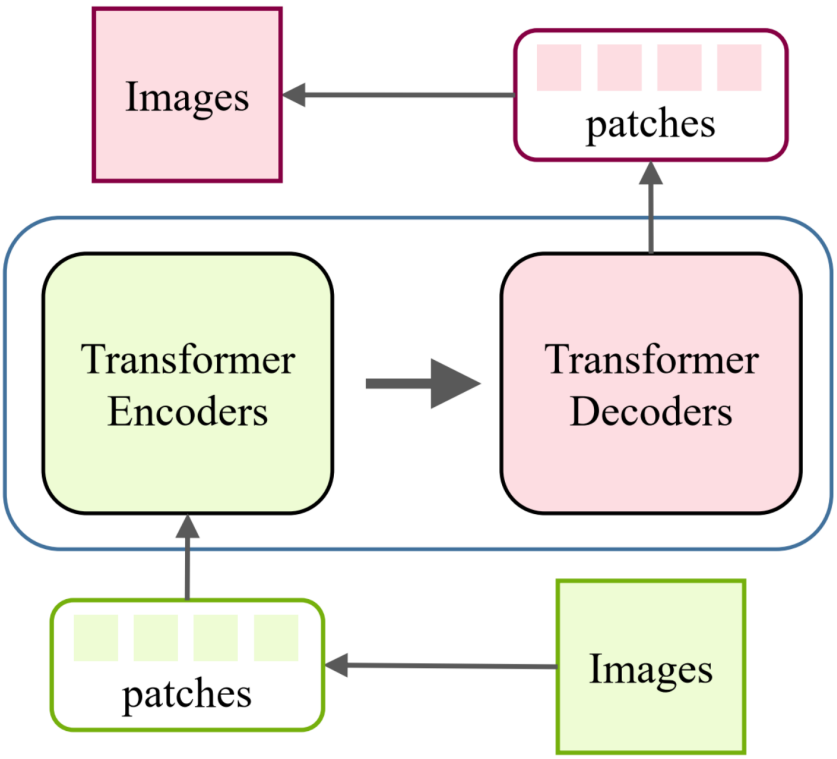


Figure 3.3: The structure of ViT in image processing (the image from (Han *et al.*, 2022)).

A simple way to apply ViT model for the image generation task is to directly change the structure from CNNs to vision transformers. Jiang *et al.* (Jiang *et al.*, 2021) proposed TransGAN based on GAN with the ViT architecture. Different from classification or detection tasks, the outputs of the model are images. Figure 3.3 shows using transformers for generation task. The images are first encoded to a sequence of image patches and the ViT encoder takes the sequence as input, enabling the decoder to successfully generate desired images. The GAN-based models directly learn a

decoder to generate image patches using linear projection. On the other hand, the transformer-based models train an auto-encoder for images and use an auto-regression ViT model to predict the image patches. A meaningful application for future research is to design different architectures for various image processing tasks. In this paper, we extend the ViT to anomaly detection in videos by integrating autoencoder in adversarial learning for anomaly detection.

3.4 System Design

To detect the events of interest on edge devices, we present a system called OED, shown in Figure 3.4. Different from the video filtering approaches based on frame-by-frame processing (Wang *et al.*, 2018a; Canel *et al.*, 2019), OED takes an entire video clip as input to detect events. Therefore the temporal information of events is best preserved. Meanwhile, processing an entire clip at once is more efficient than frame-by-frame processing. We set the clip length as 10 seconds, which is large enough to capture an event in a clip, and small enough so that frames of event detected account for a significant portion in a clip (Blair and Robertson, 2015; Doshi and Yilmaz, 2020).

3.4.1 ViT-based Autoencoder

An autoencoder neural network is an unsupervised learning model. It is trained to set the output equal to the input Ng *et al.* (2011). The autoencoder is usually used to learn a compressed representation of data. An autoencoder is organized into an encoder function and a decoder function, parameterized by θ and θ'. The encoder

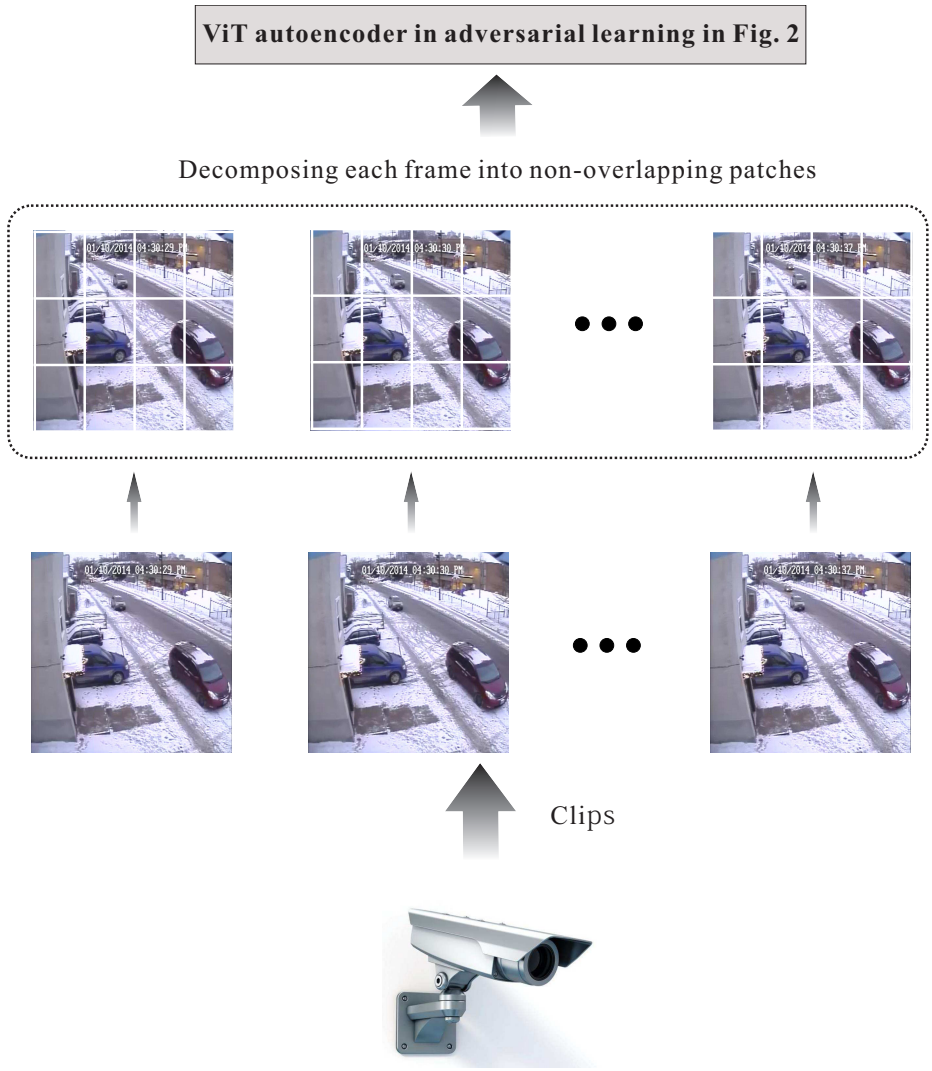


Figure 3.4: Overall system pipeline. Video clips are obtained from the camera and then we decompose each frame into non-overlapping patches (Dosovitskiy *et al.*, 2020). The patches are flattened into vectors which are fed into ViT-based autoencoder in adversarial learning (as shown in Figure 3.5) for training. Finally, abnormal events detected by the trained model from new video data are uploaded to the severs.

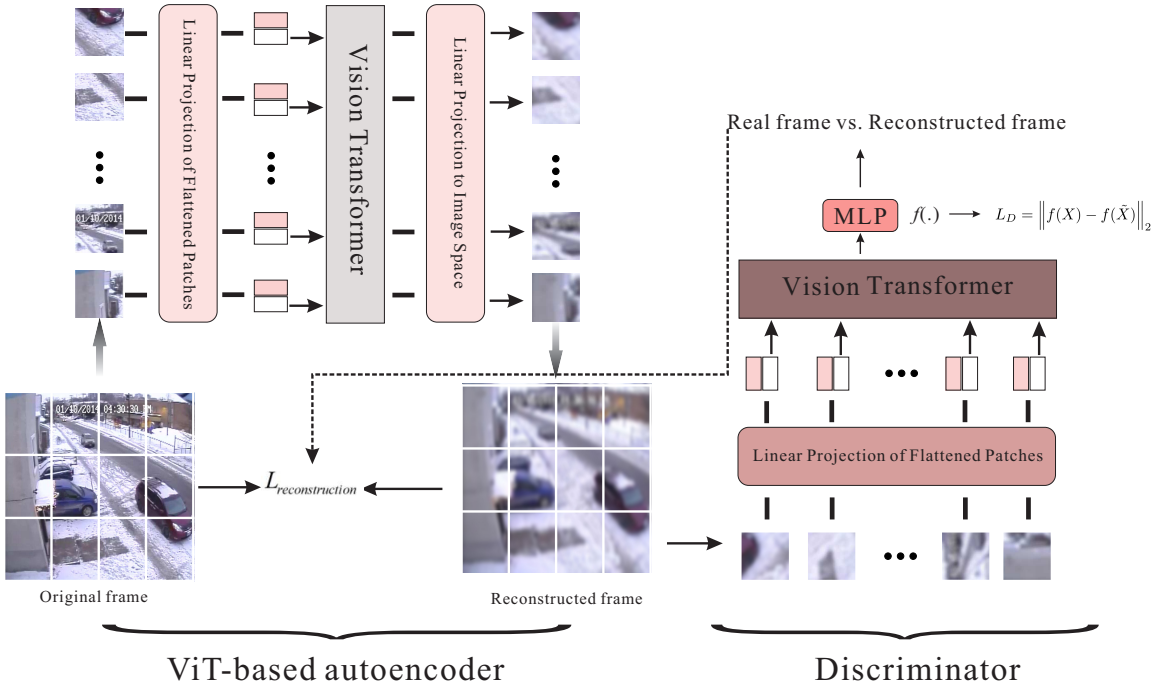


Figure 3.5: Overview of ViT-based autoencoder in adversarial learning. We take one frame as an example to show the process in training. Both the autoencoder network and the discriminator network are designed based on ViT. The two networks reinforce each other and they are in adversarial learning and unsupervised manner. The ViT-based autoencoder learns to reconstruct the normal samples and tries to fool the discriminator so that it speculates that the reconstructed sample is the original one. On the other hand, the discriminator distinguishes the original samples from reconstructed ones and is familiar with the concept of normal samples. Therefore, the discriminator will reject the reconstructed samples. The two networks play a game, and after training the ViT-based autoencoder reconstructs the normal samples with a minimum reconstruction error to successfully fool the discriminator. The two networks both learn the distribution of normal samples.

function $h = f_\theta(x)$ first maps the input vector x to a latent space representation. Then, the decoder function produces the output $r = g_{\theta'}(h)$ with the same number of the input vectors. The learning process is to minimize the following loss function by applying backpropagation with stochastic gradient descent (SGD):

$$\frac{1}{n}\sum_{i=1}^{n} L(x^i, g_{\theta'}(f_\theta(x^i))) \tag{3.4.1}$$

where each x^i represents a training sample. In general, L is defined as the mean squared error.

Convolutional autoencoder has been widely used in computer vision tasks (An and Cho, 2015; Li and He, 2019; Liu *et al.*, 2018; Doshi and Yilmaz, 2020). However, as mentioned in the Introduction, there are a few inherent limitations of convolution operators in video tasks. Therefore, We propose ViT-based autoencoder in our architecture to capture long-range dependencies in videos and reduce the computational burden. The self-attention pattern in the ViT-based autoencoder captures both local and global long-range dependencies in videos by directly comparing feature activations at all space-time locations. This extends beyond the receptive field of traditional convolutional filters. Furthermore, unlike (Dosovitskiy *et al.*, 2020), our work is based on unsupervised learning, *i.e.*, only normal samples are required in the training stage, so there is no supervised information guiding the learning process. Hence, instead of general neural networks (*e.g.*, for classification), ViT-based autoencoder is capable to ensure the intrinsic information to be preserved, thus achieving unsupervised learning manner.

Formally, given a video clip $X \in R^{H\times W\times 3\times F}$ consisting of F frames of size $H \times W$ in 3 channels (RGB). The standard Transformer takes a 1D sequence of token

embeddings as input in the field of NLP(Dosovitskiy *et al.*, 2020). To handle 2D frames in videos, a frame is decomposed into N non-overlapping patches and the size of each patch is $P \times P$, so a frame has N patches ($N = HW/P^2$). The patches are flatten into vectors $x_{(p,t)} \in R^{3\times P^2}$, where p is spatial locations and t denotes the index over frames ($p = 1, 2, ..., N$ and $t = 1, 2, ..., F$).

Next, we linearly map each patch into an embedding vector $z^{(0)}_{(p,t)} \in R^D$ with a learnable matrix $E \in R^{D\times 3\times P^2}$ (Dosovitskiy *et al.*, 2020):

$$z^{(0)}_{(p,t)} = Ex_{(p,t)} + e^{pos}_{(p,t)} \tag{3.4.2}$$

where $e^{pos}_{(p,t)} \in R^D$ represents a learnable positional embedding for encoding the spatial and temporal position of each patch. $z^{(0)}_{(p,t)}$ ($p = 1, 2, ..., N$ and $t = 1, 2, ..., F$) denotes the input to the Transformer.

For reconstruction, unlike CNN-based autoencoder that requires encoder and expensive decoder with convolutional and transposed convolution layers, the decoder of ViT-based autoencoder is implemented with simple linear layers. The ViT-based autoencoder is trained to reconstruct the input by the output tokens from the transformer. Fig. 3.5 shows the architecture of ViT-based autoencoder. "Vision Transformer" calculates the temporal attention by comparing each patch to all the patches in the same spatial location among other frames (Bertasius *et al.*, 2021).

The objective of the ViT-based autoencoder is to reconstruct the original frame. For this task, we use the L_1-loss between the original and the reconstructed frames.

Given input video frames $Z = \{ z_1, ..., z_N \}$, ViT-based autoencoder sequentially outputs the reconstruction $Y = \{ y_1, ..., y_N \}$ where $y_t \approx z_t$, $t = 1$ to N. The reconstruction error of the Z is formulated by:

$$L_{reconstruction}(Z) = \sum_{t=1}^{n} \|z_t - y_t\|^2 \tag{3.4.3}$$

3.4.2 ViT-based autoencoder in adversarial learning

Autoencoder has the ability to represent data patterns by training with observed video data. Anomaly events to be detected usually contain eccentric or unusual data patterns, which are not expected to be faithfully reconstructed by the learned autoencoder. In other words, the learned autoencoder is able to differentiate events from the routine. So we train an autoencoder to represent the current routine, and detect events by checking reconstruction errors from the autoencoder.

However, autoencoder-based methods cannot always guarantee a larger reconstruction error for abnormalities due to the indiscriminative feature learning in the training stage (Liu *et al.*, 2018). Autoencoders are not that efficient compared to GANs in reconstructing an image. As the complexity of the images increase, it is difficult for autoencoders to obtain high-quality reconstruction and images start to get blurry (Wang *et al.*, 2018b).

To overcome the drawback of autoencoders, we propose ViT-based autoencoder in adversarial learning in our architecture. Figure 3.5 illustrates our architecture, which includes the ViT-based autoencoder network and the discriminator network. The two networks reinforce each other. The ViT-based autoencoder network refines the input and gradually injects discriminative samples into the learning process to make the

normal and abnormal samples more separable for the discriminator network. In the training stage, the training data is composed of only the normal samples. Then, the ViT-based autoencoder learns to reconstruct the normal samples and tries to fool the discriminator. Whereas, the discriminator learns to distinguish input samples from reconstructed ones. In this way, the discriminator learns merely the concept characterized by the space of normal samples, giving feedback to the ViT-based autoencoder. On the other hand, the ViT-based autoencoder learns to efficiently reconstruct the normal samples, while for abnormal samples it fails to reconstruct the input accurately. Thus, the abnormalities are detected in the inference stage. Based on this, we achieve an unsupervised learning design in our system without abnormalities in the training stage.

In addition, the GAN-style methods are always unstable (model collapse and non-convergence) due to the imbalance of capability between the generator and the discriminator in training (Chen *et al.*, 2020). ViT-based autoencoder in adversarial learning balances the autoencoder and the discriminator, leading to a more stable training process with lower training loss and high-quality reconstruction.

The discriminator network distinguishes between the input frames X and the reconstructed frames $\tilde{X}$. The structure is the same as the ViT-encoder but the parameters are different. We append a 1-hidden-layer Multi-layer Perceptron (MLP) for discriminating the real frame or the reconstructed one. This network allows the whole model to achieve an adversarial scheme. The loss function is:

$$L_D = \left\| f(X) - f(\tilde{X}) \right\|_2 \tag{3.4.4}$$

where $f(.)$ is a function that outputs an intermediate layer of the discriminator.

The objective function for ViT-based autoencoder in adversarial learning combines the two loss functions:

$$L_{overall} = w_R L_{reconstruction} + w_D L_D \tag{3.4.5}$$

where w_R and w_D are the weighting parameters. $L_{reconstruction}$ and L_D represent the two loss functions.

3.4.3 Event Detection

Once the model is trained, we compute the reconstruction error $e(x) = L_{reconstruction}(x)$ of the video data for events detection. In the runtime, each video clip is input to the trained model, and the model outputs reconstruction errors. As the model identifies the majority of events, the output indicates the normal behavior. The reconstruction error indicates the probability of the input approaching the normal behavior. A smaller reconstruction error would imply that the input is normal.

In operation, we calculate the event score using the reconstruction errors. The event score is compared with a threshold for each real-time input. An event score lower than the threshold would be considered as normalities; otherwise, it is abnormal. For the threshold selection, we use Peaks-Over-Threshold (POT) (Siffer *et al.*, 2017). Given a set of historical event scores, there are two steps: 1) POT filters out samples below a specific low quantile of the entire population, fits these samples with a Generalized Pareto Distribution (GPD), and obtains the GPD function; 2) it determines the threshold using the function and an anomalous quantile among the entire population.

We then normalize the reconstruction error of video clips in the video to calculate

event scores:

$$s(x) = \frac{e(x) - min_x e(x)}{max_x e(x)} \tag{3.4.6}$$

Event scores are used to detect events of interest and the range is [0,1]. A video clip that contains events should have a high event score. A lower event score indicates no presence of abnormal events. In real-time system, we set the $min_x e(x)$ and $max_x e(x)$ according to historical data because future video data is unknown.

3.4.4 Model Updating

The proposed OED system identifies an event based on recent observations. As an analogy, a speeding car on the highway can be detected based on the observation of regular flow in the past few minutes. Hence, the model should be able to represent the recent pattern of the video contents. We propose to use sliding window updating strategy for OED. As shown in Figure 3.6, when OED scans through the video stream for event detection, the ViT autoencoder is continuously updated using several previous observations.

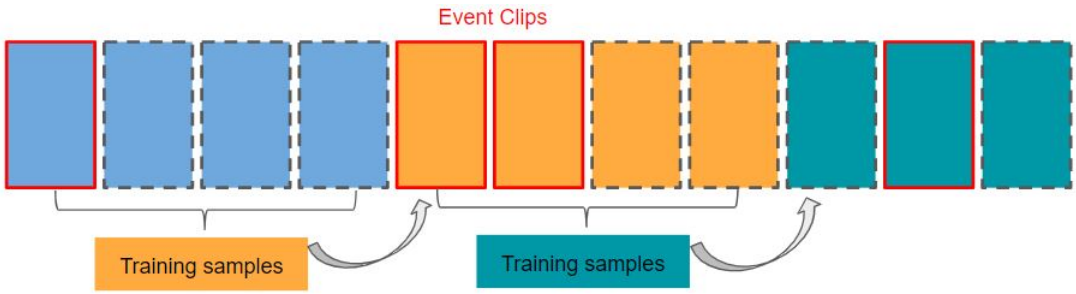


Figure 3.6: Updating strategy. Using previously observed data to re-train the model when the reconstruction error is larger than a threshold.

Specifically, given a video $V = \{X_i\}_{i=1}^{N}$, the ViT-based autoencoder is trained with initial portion of the video to gain the essential reconstruction ability. For

model updating at t-th video clip, the model is re-trained using m previous observed clips with the learning objective:

$$\frac{1}{m}\sum_{i=1}^{m} L_{reconstruction}(X_{t-i}) \tag{3.4.7}$$

For efficiency consideration, we update the model only when the reconstruction error is larger than a threshold. Since the model has already been initialized and only a small number of training samples are needed for updating, the re-training process usually takes fewer iterations to converge, which has little time cost. Moreover, the updating strategy enables OED to adapt to new trends dynamically in changing surveillance environment.

3.5 Experiments

In this section, we conduct experiments to demonstrate that OED significantly reduces bandwidth usage while maintaining good accuracy. We also evaluate the effectiveness of the updating strategy and the time cost.

3.5.1 Datasets

We collected 1630 video clips from surveillance videos in three different scenarios: neighbourhood street, subway station and industrial park. Each video clip is captured at a different time with a constant length of 10s. We annotated each video clip with the binary label $\{1, 0\}$ to indicate whether it contains events or not. Details about datasets are as follows:

Arthur Street has 180 video clips which record street views captured from a street camera deployment. There are 63 clips containing events. The dataset is downloaded from Youtube with a resolution of 640 × 360. Annotated events include walking pedestrians and passing vehicles. Background colour change is also recorded in this dataset.

Subway Exit / Entrance has 935 video clips. 358 clips contain events. This dataset consists of more than 2 hours long surveillance videos captured from a subway station with a resolution of 480 × 360. One video monitors the subway exit and the other monitors the subway entrance. The dataset is provided by (Adam *et al.*, 2008). Annotated events include: entering the entrance; coming out from the exit; waiting, talking and walking around; train coming; getting on/off the train.

Industrial Park consists of the video collected from a camera deployment in an industrial park with a resolution of 352 × 288. There are totally 515 video clips. 19 clips contain events, reflecting the typical outdoor surveillance scenario. The background in videos involves swaying tree leaves, shadows and lighting changes. Events in this dataset are very rare, which are mainly caused by bicycles, pedestrians and animals.

3.5.2 Setup

All frames are resized to 224×224 and the patch size is 16×16. OED is implemented in Python and trained on a workstation with core i7-7700k 4.20GHz processor and 32GB RAM using Ubuntu system with graphics card GeForce GTX 1080. In our experiment, after training, we only use a quad-core Intel CPU that is representative of an edge node. All experiments are conducted using TensorFlow (Abadi *et al.*,

2016).

	Recall	Precision	F1 Score	Uploaded Fraction	Events Fraction
Arthur Street	0.91	0.60	0.72	0.56	0.37
Subway Exit	0.89	0.68	0.77	0.60	0.46
Subway Entrance	0.91	0.82	0.86	0.68	0.62
Industrial Park	0.79	0.41	0.54	0.07	0.04

Table 3.1: Accuracy and uploaded clip fraction on three datasets

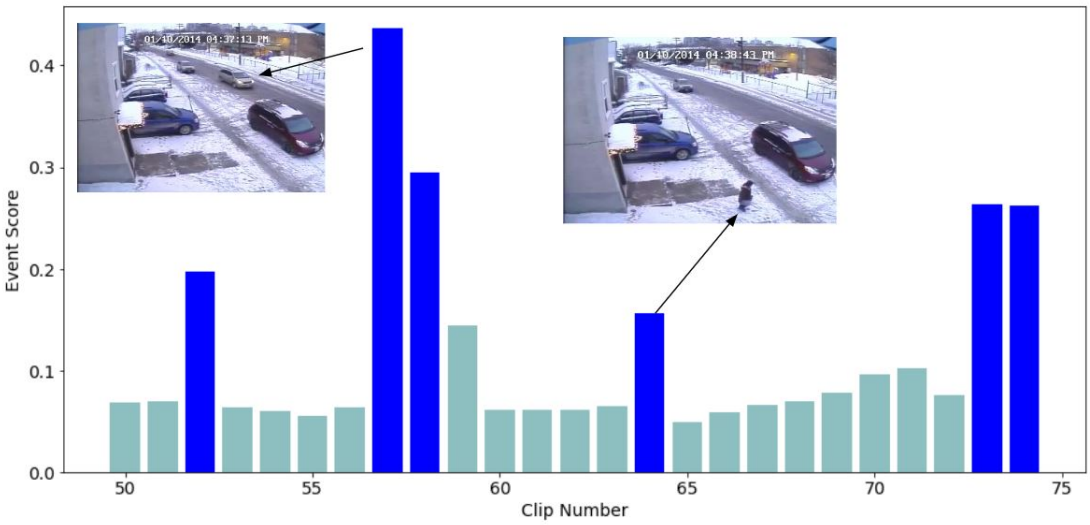


Figure 3.7: Event scores of a video sequence from Arthur Street dataset. Blue parts is detected event clips at a chosen threshold. The left image shows "passing vehicle" and the right image shows "walking kid".

3.5.3 System Performance

Performance Indicators

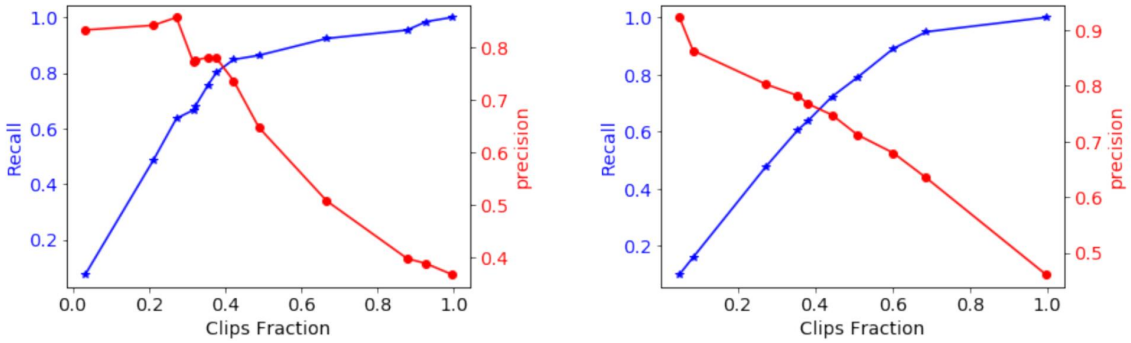
To evaluate the performance of event detection and bandwidth saving, we adopt the similar evaluation strategies in (Wang *et al.*, 2018a; Canel *et al.*, 2019). The classic

computer vision task metrics, *precision*, *recall* and *F1 score* are used to measure the event detection accuracy. *Precision* determines what fraction of bandwidth is used to send data with events. A poor precision indicates detection results contain too many false positives, which result in overload for bandwidth. *Recall* is the fraction of events that are successfully retrieved. Since we do not want to miss important information such as suspicious behaviors, a practicable approach is to achieve a higher recall and a slightly lower precision. *F1 score* is the harmonic mean of the precision and recall, using as an overall measure of the accuracy. For bandwidth saving, we are interested in the average bandwidth demand, which is calculated as the total volume of uploaded data divided by the video duration.

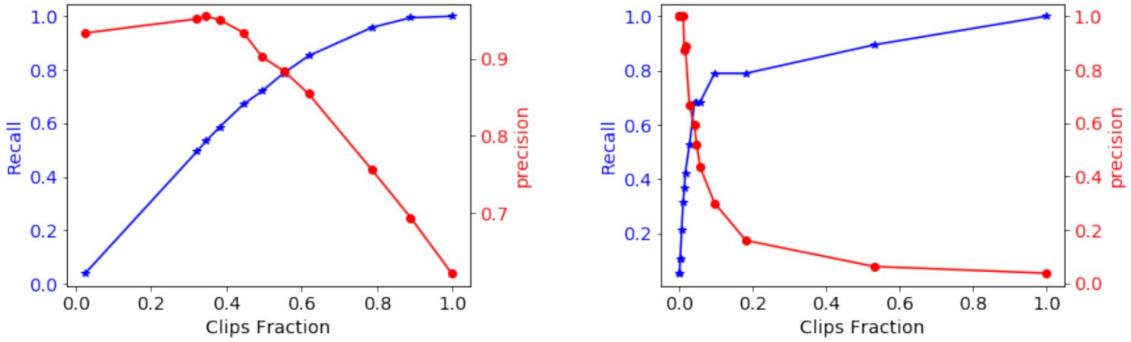
Events Detection Accuracy

The output of OED is the event score of the current video clip, as shown in Figure 3.7. Up to 90%) can be detected. Figure 3.8 relates the uploaded clips fraction with recall and precision on different surveillance scenarios. This figure also shows the trade-off between event recall and precision. It allows us to evaluate how many fractions of clips need to be uploaded to achieve good accuracy.

Events occur very frequently in Arthur Street and Subway Exit/Entrance datasets. There are about half of the clips containing events. When uploading about 60% of clips, OED is able to achieve a high recall while keeping a good precision. The majority of events are detected, and most of the uploaded clips are true positives. For Industrial Park dataset, recall is good even only upload 7% of video clips. It indicates bandwidth is saved impressively. But achieving a better recall may lead to a slightly lower precision. Table 3.1 provides quantitative results on events detection



(a) Arthur Street (b) Subway Exit



(c) Subway Entrance (d) Industrial Park

Figure 3.8: Relation between events recall, precision and uploaded clips fraction.

accuracy. Up to 90% of events are correctly detected, while only uploading 7% to 68% of the video clips depending on different surveillance scenarios.

ViT-based autoencoder in adversarial learning enables our system to capture the normal data distribution as much as possible in the training stage, thus enlarging the reconstruction errors for abnormal samples in the inference stage. Therefore, OED achievers adversarial mechanism, making high-quality reconstruction based on ViT autoencoder. We compare OED with and without adversarial learning. For OED without adversarial learning, we only use ViT autoencoder. As shown in Table 3.2, OED without adversarial learning fails to achieve good performance because autoencoder without adversarial training results in low reconstruction errors for abnormal data. We observe that the *recall* is further improved (*F1 score* from 0.81 to 0.86). Therefore, we propose adversarial learning in our architecture to enlarge the recon-

Methods	Recall	Precision	F1 Score
Without adversarial learning	0.87	0.77	0.81
With adversarial learning	0.91	0.82	0.86

Table 3.2: Recall, precision and F1 scores without and with adversarial learning on "Subway Entrance" dataset

struction errors for abnormalities in the inference stage. Benefiting from adversarial learning, OED achieves better performance than that without adversarial learning. This experiment validates the effectiveness and necessity of adversarial learning for capture the normal data distribution, improving the performance of our system.

Furthermore, we show the comparison of different methods and the performance evaluation in Table 3.3. Traditionally, the features of video clips are extracted from

Methods	Recall	Precicion	F1 Score
MobileNet with (Srivastava *et al.*, 2015)	0.85	0.65	0.74
MobileNet with VAE	0.82	0.61	0.70
Ours	0.89	0.68	0.77

Table 3.3: Recall, precision and F1 scores for different methods on "Subway Exit" dataset

the higher layer of CNNs. Because our system is real-time event detection on edge. Therefore, we also use MobileNet (Howard *et al.*, 2017) to extract features from each frame. Compared with other complicated models, MobileNet has fewer model parameters and the feature is extracted in the terminal devices for edge computing. Each frame feature is a 1024-dimensional vector and extracted from the fully connected layer of pre-trained MobileNet. After that, feature vectors are used as the inputs. The method based on Long Short-Term Memory (LSTM) autoencoder (Srivastava *et al.*, 2015) with MobileNet learns spatio-temporal normal patterns for detection. However, different from CNNs that is capable of convolution (weight-sharing and local-connectivity), ViT relies on globally-contextualized representation, replacing altogether the convolution operator with self-attention patterns and demonstrating advantages in non-local contextual dependencies. Therefore, ViT captures long-range dependencies that extend beyond the receptive field among video frames.

Variational Auto Encoder (VAE) (An and Cho, 2015) is a deep generative model for anomaly detection. An *et al.* proposed to utilize the generative characteristics of the variational autoencoder for deriving the reconstruction of the data to detect anomalies. However, the main problem is that the fixed parametrization of VAE is oversimplified compared to the true complex distribution of real-world data. This is not a problem of ViT-based autoencoder because it is forced to make the reconstructed

samples to follow the distribution of true data and the model is more hierarchical.

Saving Bandwidth

We demonstrate that OED achieves the goal of saving network bandwidth while keeping a good accuracy. Specifically, performing filtering with OED can reduce average bandwidth consumption by 2 to 15 times compared with uploading all video data to servers. In our evaluation, average bandwidth is the total uploaded video data divided by the original video duration. Figure 3.9 shows results on five video data. In this figure, we evaluate two techniques for uploading video from the edge: run OED on the edge, and then select event clips to upload; Directly upload full video data.

The amounts of bandwidth saved also depend on the rarity of events. The lower the frequency of events in the video, the more bandwidth OED will save. Experimental results show that filtering on Industrial Park saves more bandwidth than the other surveillance scenarios.

3.5.4 Evaluation of Updating Strategy

OED uses an updating strategy for event detection. The model is updated using previous observations when reconstruction error is larger than a threshold. In this section, we evaluate the effectiveness of the updating strategy, which is designed to capture the change of routines in the video. We compare our approach with one using offline initially trained model throughout the entire video. Specifically, the approach without the updating strategy only trains the model using the first 16 clips of video before the detection process.

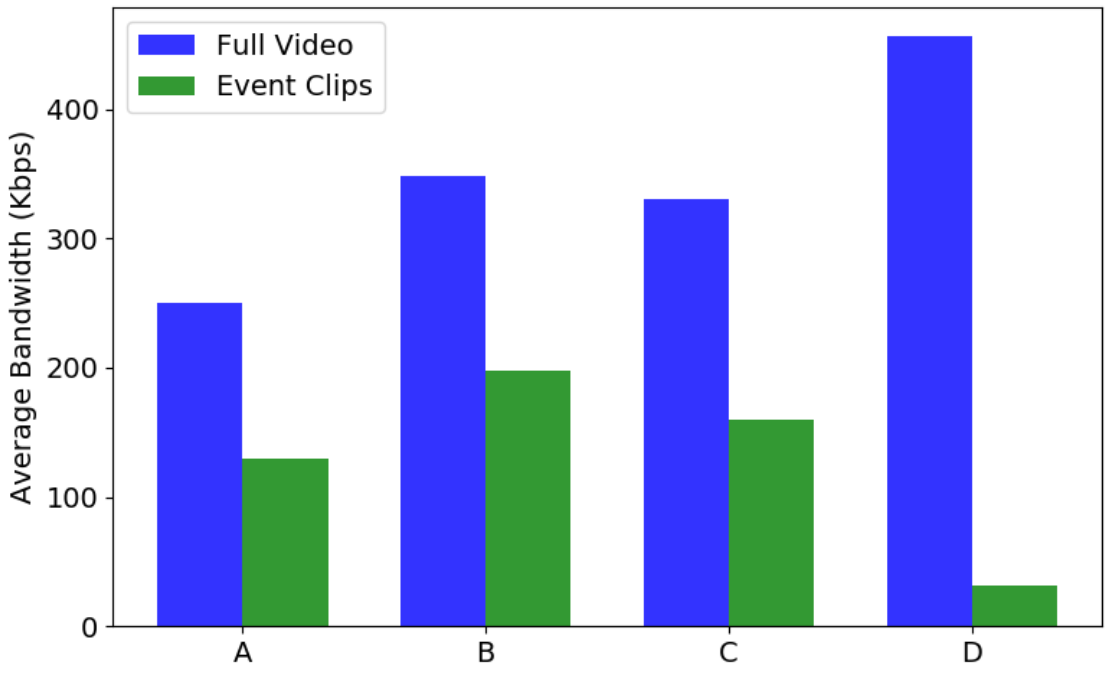


Figure 3.9: Bandwidth saving of OED with same accuracy on Table 3.1. A: Arthur Street; B: Subway Exit; C: Subway Entrance; D: Industrial Park.

	Recall (Update / No Update)	Precision (Update / No Update)
Arthur Street	0.91 / 0.57	0.60 / 0.54
Subway Exit	0.89 / 0.56	0.72 / 0.42
Subway Entrance	0.91 / 0.65	0.73 / 0.55
Industrial Park	0.79 / 0.41	0.41 / 0.30

Table 3.4: Comparison of events detection accuracy: updating vs. no updating.

Figure 3.10 shows clip reconstruction errors with the above two approaches on Arthur Street dataset. We demonstrate that the approach with the updating strategy effectively adapts to environmental changes in the video. At 85-*th* video clip, the background color changes to black and white, which causes a large reconstruction error on both approaches. The approach with updating strategy adapts to the new trends immediately, while the other one fails to detect events thereafter. Besides, the model without updating strategy cannot avoid concept drift in a gradually changing surveillance environment.

We also compare the event detection accuracy of the two approaches when setting the same uploaded clip fraction in Table 3.4. The model with updating strategy performs favorably against the other approach in all datasets, showing its effectiveness in event detection.

3.5.5 Time Cost

Since our approach is edge-based, only limited computational resources are available. We evaluate the efficiency of OED by observing the processing time. Table 3.5 shows the processing time of OED. T_{video} is the duration of the original video. T_1 is the ViT-based autoencoder time. T_2 is the time spent on initial training. T_3 represents

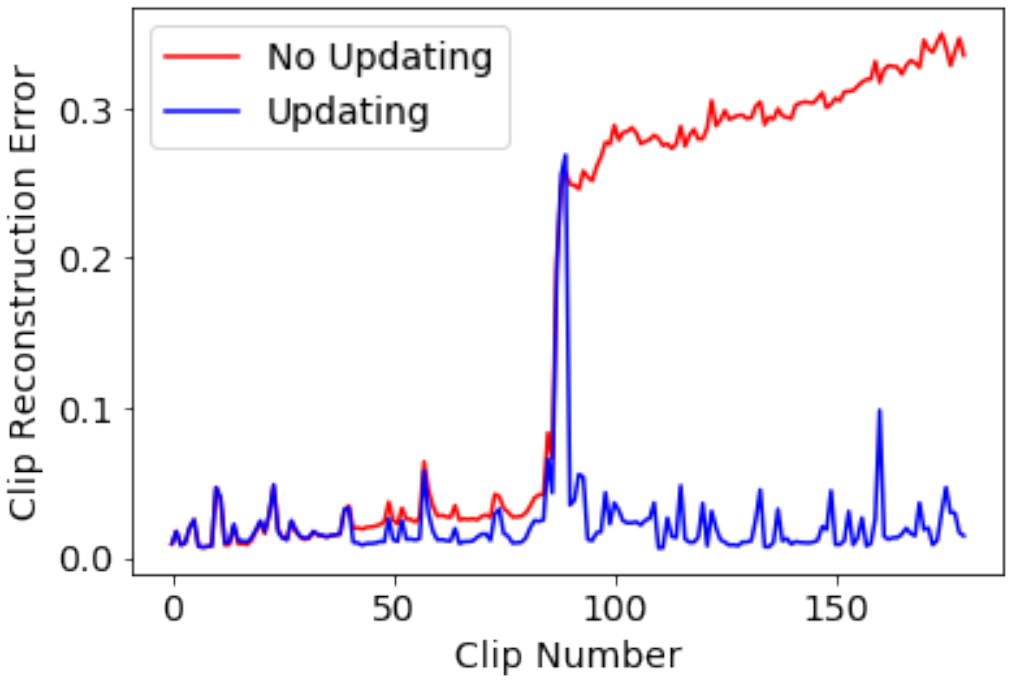


Figure 3.10: Clip reconstruction errors on Arthur Street dataset with the updating strategy and without the updating strategy. The background color in the video changes at the clip number of 85. Higher values indicate events in clips.

the time for model updating and event detection. $T = T_1 + T_2 + T_3$ is the total processing time. From the results in the table, we can get the following observations: (1) Vision transformer T_1 is the most time-consuming part. The complexity of T_1 appears to be linear with regard to the video duration. (2) Initial training time T_2 is typically short because only small number of clips are used for training. (3) The updating and event detection time T_3 is nonlinear with the video duration because the updating frequency is depending on surveillance scenarios. (4) For all scenarios, the total processing time T is much smaller than the video length T_{video}. At run time, we run our system on a CPU that is representative of an edge node and the processing time is about 85fps (frames per second) on the surveillance scenarios. That means OED qualifies for real-time task processing.

	T_1	T_2	T_3	T	T_{video}
Arthur Street	5.16	2.71	2.96	10.83	30.50
Subway Exit	8.44	1.64	6.09	16.17	43.27
Subway Entrance	19.48	3.48	14.71	37.64	96.17
Industrial Park	14.55	2.58	7.76	24.89	85.92

Table 3.5: Processing time (minutes) of OED.

3.5.6 Visualization

We visualize the detection results of our approach in this section. Figure 3.11 shows some results on Arthur Street dataset. Most detected events are caused by moving objects of interest in a quiet environment, such as pedestrians walking down the sidewalk (Figure 3.11(a)), the fast passing vehicle (Figure 3.11(b)), adults and children going out of the building (Figure 3.11(c)). It also illustrates that background color

changes in the video do not affect event detection (Figure 3.11(c)). Several missing events are due to small or distant objects such as a man in the distance (Figure 3.11(d)).

(a) Detected event: pedestrians

(b) Detected event: passing vehicle

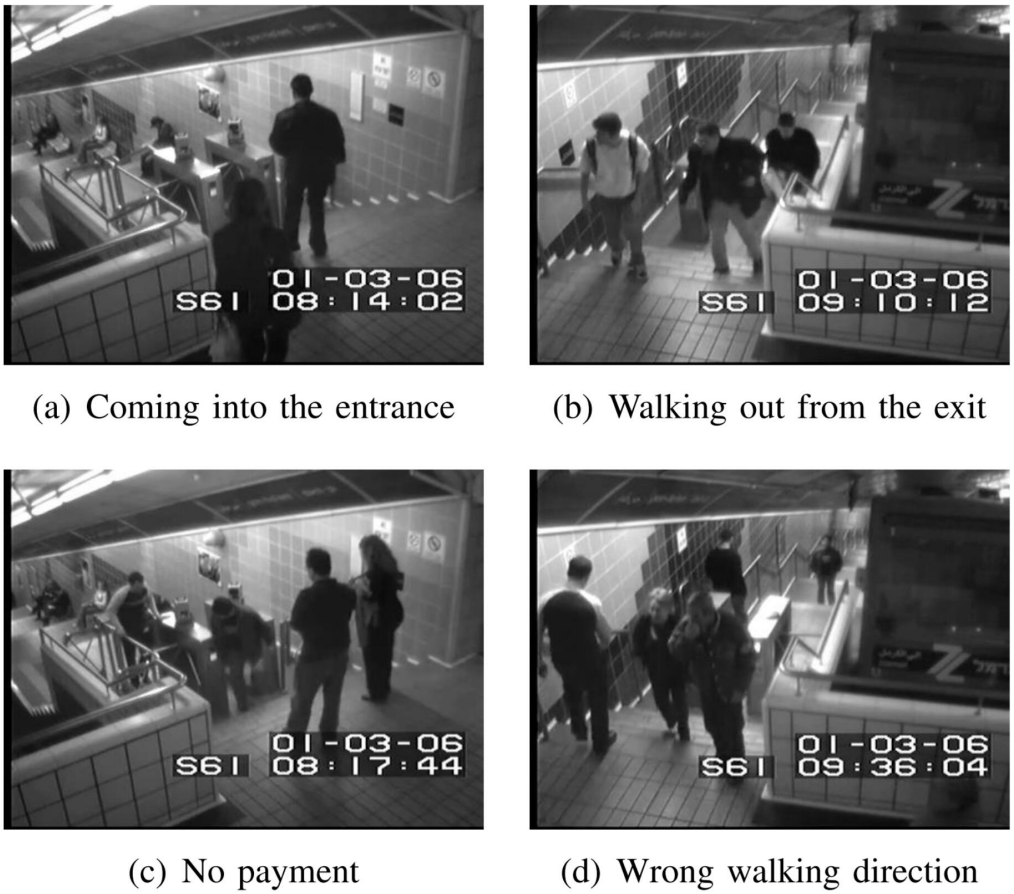
(c) Detected event: adult and child going out of the building

(d) Missing event: a man in the distance

Figure 3.11: Visualization of events in Arthur Street dataset. (c) and (d) are captured after background color changes.

Figure 3.12 displays the results in Subway Exit/Entrance dataset. Most of the detected events are typical daily happening activities: walking to the entrance (Figure 3.12(a)), walking out from the exit (Figure 5(b)). We also illustrate detected unusual activities such as without payment (Figure 3.12(c)) and wrong walking direction (Figure 3.12(d)). These two unusual events have higher event scores (0.53 and 0.6)

than other detected ones (mostly between 0.1 to 0.3).



(a) Coming into the entrance (b) Walking out from the exit

(c) No payment (d) Wrong walking direction

Figure 3.12: Visualization of events in Subway Exit/Entrance dataset.

For Industrial Park dataset, events are very rare. We visualize some detected events in Figure 3.13: walking around (Figure 3.13(a)) and riding the bicycle (Figure 3.13(b)). False detected events are mainly resulted from nuisance sources such as lights and shadows.

(a) walking around (b) riding the bicycle

Figure 3.13: Visualization of events in Industrial Park dataset.

3.6 Conclusion

We studied the problem of event detection in surveillance videos. We present an efficient and effective system OED that only uploads event clips to servers to save bandwidth. OED learns video patterns from recently observed data based on ViT-based autoencoder in an unsupervised manner, so it gains the ability to differentiate events from routine. ViT-based autoencoder system is trained on normal samples only and its parameters are only suitable for reconstructing normal ones. In the inference stage, for a given anomaly sample, the model poorly reconstructs the input sample and the reconstruction error would be high. The high error indicates the input sample is abnormal. Based on this, we achieve unsupervised learning design in our system without abnormal samples in the training stage. The updating strategy enables OED to adapt to new trends in videos dynamically. Evaluations demonstrate that OED reduces bandwidth usage significantly while maintaining a good event detection accuracy.

Chapter 4

Conclusion and Future Work

4.1 Conclusion

Video data is challenging to represent and model due to its high dimensionality, noise, and a huge variety of events among frames. Anomalies are also highly contextual. The core problem for anomaly detection for videos is learning the discriminative video descriptor with only normal samples in the training stage. The descriptor should preserve both spatial and temporal information thus can be easily detected. To generate the discriminative video descriptor, the following questions should be answered. 1. How to preserve the entire temporal information of videos with various lengths. 2. How to capture the distribution of training data. 3. How to detect anomalies using video descriptors. In this thesis, we try to answer these questions and propose effective methods that preserve the temporal information of the entire video. Meanwhile, for anomaly detection for videos, the end-to-end supervised learning methods are only applicable to labeled video data where events of interest are clearly defined. Furthermore, the cost of labeling every type of event is extremely high. Even so,

it is not possible to cover every past and future events. The recorded video data is likely not long enough to capture all types of activities, especially abnormal events which rarely or never occurred. Our approaches achieve unsupervised learning based on reconstruction and adversarial learning.

There are two types of anomaly detection for videos: anomalous video detection and anomalous event detection in videos. Anomalous video detection is related to organizing video resources, which enables video-sharing platforms or public sectors to focus on the specific video contents and filter unconsidered ones; anomalous event detection in videos is applicable in the scenarios like monitoring in smart homes, smart cities and Internet of Things, which allows surveillance cameras to upload events of interest so as to suppress the network traffic and reduce storage space in the cloud.

We first propose and develop a LSTM-autoencoder-based adversarial learning system, called VidAnomaly. Rather than directly processing every video frame or extracting feature vectors from video, we introduce dynamic image to summarize the video contents into a 2D image that encodes the temporal evolution and preserve spatial information of the video content. To keep the temporal information as much as possible, we segregate the video into segments and extract multiple dynamic images from them. Without the abnormal samples in the training stage, we propose LSTM-autoencoder-based adversarial learning network. The input in our task is the sequence and LSTM-autoencoder yields the low-dimension latent representation and reconstructs the input sequence for the discriminator network to achieve adversarial learning. The novelty of LSTM-autoencoder-based adversarial learning network is to add an additional LSTM-encoder network that encoders the reconstructed sequence to obtain the latent representation. The difference between the two latent

representations is further enlarged when the input is abnormal samples. LSTM-autoencoder-based adversarial learning network is trained on normal samples only and its parameter is only suitable for reconstructing normal samples. In the inference stage, for a given abnormal sample, the model poorly reconstructs the input sample and the reconstruction error would be high. Based on this, we achieve anomalous video detection without abnormal ones in the training stage.

For anomalous event detection in videos, we present an efficient and effective system OED that only uploads event clips to servers to save bandwidth. OED learns video patterns from recently observed data based on ViT-based autoencoder in an unsupervised manner, so it gains the ability to differentiate events from routine. The updating strategy enables OED to adapt to new trends in videos dynamically.

4.2 Future Work

The first future work is evaluating the MLP-Mixer embedding into our architecture. Much effort has been made recently to attempt to replace self-attention with Multiple Layer Perceptron (MLP). This is because the fully-connected layer is also able to model the long-range dependencies. The MLP-Mixer architecture, a competitive but conceptually and technically simple alternative, indicates that self-attention may not be a must-have. MLP-Mixer is based entirely on MLPs and relies only on basic matrix multiplication routines. Different from the convolution operation in CNNs, MLPMixer repeats implementation of Layer Norm and MLP either in spatial locations or feature channels. The long-term dependencies of the images are covered. The MLP-Mixer thus has great potential to be beneficial for capturing the global dependencies.

Bibliography

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., *et al.* (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

Abe, N., Zadrozny, B., and Langford, J. (2006). Outlier detection by active learning. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 504–509.

Adam, A., Rivlin, E., Shimshoni, I., and Reinitz, D. (2008). Robust real-time unusual event detection using multiple fixed-location monitors. *IEEE transactions on pattern analysis and machine intelligence*, **30**(3), 555–560.

Akcay, S., Atapour-Abarghouei, A., and Breckon, T. P. (2018). Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian conference on computer vision*, pages 622–637. Springer.

An, J. and Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, **2**(1), 1–18.

Ashfaq, R. A. R., Wang, X.-Z., Huang, J. Z., Abbas, H., and He, Y.-L. (2017).

Fuzziness based semi-supervised learning approach for intrusion detection system. *Information Sciences*, **378**, 484–497.

Barnett, T., Jain, S., Andra, U., and Khurana, T. (2018). Cisco visual networking index (vni) complete forecast update, 2017–2022. *Americas/EMEAR Cisco Knowledge Network (CKN) Presentation*.

Barrett, D. (2013). One surveillance camera for every 11 people in britain, says cctv survey.

Beal, J., Kim, E., Tzeng, E., Park, D. H., Zhai, A., and Kislyuk, D. (2020). Toward transformer-based object detection. *arXiv preprint arXiv:2012.09958*.

Bertasius, G., Wang, H., and Torresani, L. (2021). Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*.

Bilen, H., Fernando, B., Gavves, E., Vedaldi, A., and Gould, S. (2016). Dynamic image networks for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3034–3042.

Blair, C. G. and Robertson, N. M. (2015). Video anomaly detection in real time on a power-aware heterogeneous platform. *IEEE Transactions on Circuits and Systems for Video Technology*, **26**(11), 2109–2122.

Cai, S., Zuo, W., Davis, L. S., and Zhang, L. (2018). Weakly-supervised video summarization using variational encoder-decoder and web prior. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 184–200.

Camacho, J., Pérez-Villegas, A., García-Teodoro, P., and Maciá-Fernández, G.

(2016). Pca-based multivariate statistical network monitoring for anomaly detection. *Computers & Security*, **59**, 118–137.

Campos, G. O., Zimek, A., Sander, J., Campello, R. J., Micenková, B., Schubert, E., Assent, I., and Houle, M. E. (2016). On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. *Data mining and knowledge discovery*, **30**(4), 891–927.

Canel, C., Kim, T., Zhou, G., Li, C., Lim, H., Andersen, D. G., Kaminsky, M., and Dulloor, S. R. (2019). Scaling video analytics on constrained edge nodes. *arXiv preprint arXiv:1905.13536*.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer.

Chang, J., Wang, L., Meng, G., Xiang, S., and Pan, C. (2017). Deep adaptive image clustering. In *Proceedings of the IEEE international conference on computer vision*, pages 5879–5887.

Chen, C., Liu, J., Xie, Y., Ban, Y. X., Wu, C., Tao, Y., and Song, H. (2020). Latent regularized generative dual adversarial network for abnormal detection. In *IJCAI*, pages 760–766.

Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., and Gao, W. (2021). Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12299–12310.

Chen, T. Y.-H., Ravindranath, L., Deng, S., Bahl, P., and Balakrishnan, H. (2015). Glimpse: Continuous, real-time object recognition on mobile devices. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 155–168. ACM.

Chen, Y., He, W., Hua, Y., and Wang, W. (2016). Compoundeyes: Near-duplicate detection in large scale online video systems in the cloud. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE.

Chong, Y. S. and Tay, Y. H. (2017). Abnormal event detection in videos using spatiotemporal autoencoder. In *International Symposium on Neural Networks*, pages 189–196. Springer.

Cordonnier, J.-B., Loukas, A., and Jaggi, M. (2019). On the relationship between self-attention and convolutional layers. In *International Conference on Learning Representations*.

Culurciello, E. and Canziani, A. (2017). e-Lab video data set. `https://engineering.purdue.edu/elab/eVDS/`.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Donahue, J., Krähenbühl, P., and Darrell, T. (2016). Adversarial feature learning. *arXiv preprint arXiv:1605.09782*.

Doshi, K. and Yilmaz, Y. (2020). Fast unsupervised anomaly detection in traffic videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 624–625.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., *et al.* (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Fernando, B., Gavves, E., Oramas, J. M., Ghodrati, A., and Tuytelaars, T. (2015). Modeling video evolution for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5378–5387.

Fernando, B., Anderson, P., Hutter, M., and Gould, S. (2016a). Discriminative hierarchical rank pooling for activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1924–1932.

Fernando, B., Gavves, E., Oramas, J., Ghodrati, A., and Tuytelaars, T. (2016b). Rank pooling for action recognition. *IEEE transactions on pattern analysis and machine intelligence*, **39**(4), 773–787.

Gaddam, S. R., Phoha, V. V., and Balagani, K. S. (2007). K-means+ id3: A novel method for supervised anomaly detection by cascading k-means clustering and id3 decision tree learning methods. *IEEE transactions on knowledge and data engineering*, **19**(3), 345–354.

Gardner, A. B., Krieger, A. M., Vachtsevanos, G., Litt, B., and Kaelbing, L. P. (2006). One-class novelty detection for seizure analysis from intracranial eeg. *Journal of Machine Learning Research*, **7**(6).

Gavrilyuk, K., Sanford, R., Javan, M., and Snoek, C. G. (2020). Actor-transformers for group activity recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 839–848.

Ghasedi, K., Wang, X., Deng, C., and Huang, H. (2019). Balanced self-paced learning for generative adversarial clustering network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4391–4400.

Girdhar, R., Carreira, J., Doersch, C., and Zisserman, A. (2019). Video action transformer network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–253.

Gkioxari, G. and Malik, J. (2015). Finding action tubes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 759–768.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, **27**.

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y., *et al.* (2022). A survey on vision transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Hasan, M., Choi, J., Neumann, J., Roy-Chowdhury, A. K., and Davis, L. S. (2016).

Learning temporal regularity in video sequences. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 733–742.

Hinami, R., Mei, T., and Satoh, S. (2017). Joint detection and recounting of abnormal events by learning deep generic knowledge. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3619–3627.

Hoai, M. and Zisserman, A. (2014). Improving human action recognition using score distribution and ranking. In *Asian conference on computer vision*, pages 3–20. Springer.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Ionescu, R. T., Smeureanu, S., Popescu, M., and Alexe, B. (2019). Detecting abnormal events in video using narrowed normality clusters. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1951–1960. IEEE.

Ji, S., Xu, W., Yang, M., and Yu, K. (2012). 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, **35**(1), 221–231.

Jiang, Y., Chang, S., and Wang, Z. (2021). Transgan: Two pure transformers can make one strong gan, and that can scale up. *Advances in Neural Information Processing Systems*, **34**.

Jiang, Y.-G., Ye, G., Chang, S.-F., Ellis, D., and Loui, A. C. (2011). Consumer video understanding: A benchmark database and an evaluation of human and

machine performance. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, pages 1–8.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., and Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732.

Khan, S. S. and Madden, M. G. (2014). One-class classification: taxonomy of study and review of techniques. *The Knowledge Engineering Review*, **29**(3), 345–374.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Welling, M. (2014). Stochastic gradient vb and the variational auto-encoder. In *Second International Conference on Learning Representations, ICLR*, volume 19, page 121.

Kriegel, H.-P., Schubert, M., and Zimek, A. (2008). Angle-based outlier detection in high-dimensional data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 444–452.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, **25**, 1097–1105.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, **1**(4), 541–551.

Lee, J., Reade, W., Sukthankar, R., Toderici, G., *et al.* (2018). The 2nd youtube-8m large-scale video understanding challenge. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0.

Lee, K., Chang, H., Jiang, L., Zhang, H., Tu, Z., and Liu, C. (2021). Vitgan: Training gans with vision transformers. *arXiv preprint arXiv:2107.04589*.

Li, S. and He, W. (2019). Vidanomaly: Lstm-autoencoder-based adversarial learning for one-class video classification with multiple dynamic images. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 2881–2890. IEEE.

Liu, W., Luo, W., Lian, D., and Gao, S. (2018). Future frame prediction for anomaly detection–a new baseline. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6536–6545.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, **60**(2), 91–110.

Mahasseni, B., Lam, M., and Todorovic, S. (2017). Unsupervised video summarization with adversarial lstm networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 202–211.

Ng, A. *et al.* (2011). Sparse autoencoder. *CS294A Lecture notes*, **72**(2011), 1–19.

Pakha, C., Chowdhery, A., and Jiang, J. (2018). Reinventing video streaming for distributed vision analytics. In *10th {USENIX} Workshop on Hot Topics in Cloud Computing (HotCloud 18)*.

Ren, M., Zeng, W., Yang, B., and Urtasun, R. (2018). Learning to reweight examples

for robust deep learning. In *International Conference on Machine Learning*, pages 4334–4343. PMLR.

Ryoo, M. S., Rothrock, B., and Matthies, L. (2015). Pooled motion features for first-person videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 896–904.

Sabokrou, M., Fathy, M., and Hoseini, M. (2016). Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder. *Electronics Letters*, **52**(13), 1122–1124.

Sabokrou, M., Khalooei, M., Fathy, M., and Adeli, E. (2018). Adversarially learned one-class classifier for novelty detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3379–3388.

Sakurada, M. and Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, pages 4–11.

Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., and Langs, G. (2017). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, pages 146–157. Springer.

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, **404**, 132306.

Siffer, A., Fouque, P.-A., Termier, A., and Largouet, C. (2017). Anomaly detection in streams with extreme value theory. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1067–1075.

Simonyan, K. and Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*.

Smola, A. J. and Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, **14**(3), 199–222.

Srivastava, N., Mansimov, E., and Salakhudinov, R. (2015). Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852.

Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. (2015). Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wang, J., Feng, Z., Chen, Z., George, S., Bala, M., Pillai, P., Yang, S.-W., and Satyanarayanan, M. (2018a). Bandwidth-efficient live video analytics for drones via edge computing. In *2018 IEEE/ACM Symposium on Edge Computing (SEC)*, pages 159–173. IEEE.

Wang, J., Zhou, W., Tang, J., Fu, Z., Tian, Q., and Li, H. (2018b). Unregularized auto-encoder with generative adversarial networks for image generation. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 709–717.

Wang, X., Girshick, R., Gupta, A., and He, K. (2018c). Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803.

Weissenborn, D., Täckström, O., and Uszkoreit, J. (2019). Scaling autoregressive video models. *arXiv preprint arXiv:1906.02634*.

Xiao, Y., Jia, Y., Liu, C., Cheng, X., Yu, J., and Lv, W. (2019). Edge computing security: State of the art and challenges. *Proceedings of the IEEE*, **107**(8), 1608–1631.

Xu, D., Ricci, E., Yan, Y., Song, J., and Sebe, N. (2015). Learning deep representations of appearance and motion for anomalous event detection. *arXiv preprint arXiv:1510.01553*.

Yang, H., Wang, B., Lin, S., Wipf, D., Guo, M., and Guo, B. (2015). Unsupervised extraction of video highlights via robust recurrent auto-encoders. In *Proceedings of the IEEE international conference on computer vision*, pages 4633–4641.

Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., and Toderici, G. (2015). Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702.

Zenati, H., Foo, C. S., Lecouat, B., Manek, G., and Chandrasekhar, V. R. (2018). Efficient gan-based anomaly detection. *arXiv preprint arXiv:1802.06222*.

Zhai, S., Cheng, Y., Lu, W., and Zhang, Z. (2016). Deep structured energy based models for anomaly detection. In *International Conference on Machine Learning*, pages 1100–1109. PMLR.

Zhang, T., Chowdhery, A., Bahl, P. V., Jamieson, K., and Banerjee, S. (2015). The design and implementation of a wireless video surveillance system. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pages 426–438. ACM.

Zhang, Y., Liang, X., Zhang, D., Tan, M., and Xing, E. P. (2018). Unsupervised object-level video summarization with online motion auto-encoder. *Pattern Recognition Letters*.

Zhao, B., Fei-Fei, L., and Xing, E. P. (2011). Online detection of unusual events in videos via dynamic sparse coding. In *CVPR 2011*, pages 3313–3320. IEEE.

Zhao, H., Jia, J., and Koltun, V. (2020). Exploring self-attention for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10076–10085.

Zhao, Y., Deng, B., Shen, C., Liu, Y., Lu, H., and Hua, X.-S. (2017). Spatio-temporal autoencoder for video anomaly detection. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1933–1941. ACM.