Dissertation for Doctor of Philosophy

Generative Models for Anomaly Detection and Its Applications

Jongmin Yu

School of Electrical Engineering and Computer Science

Gwangju Institute of Science and Technology

2020

PhD/EC Jongmin Yu. Generative Models for Anomaly Detection
and Its Applications. School of Electrical Engineering and Computer Science.
2020. 161p. Advisor: Prof. Moongu Jeon.

Abstract

Anomaly detection is a process for distinguishing the observations that differ in some respect from the observations that the model is trained on. Anomaly detection is one of the fundamental requirements of a good classification or identification system since sometimes the test data contains observations that were not known at the training time. In other words, the anomaly class is often is not presented during the training phase or not well defined. In light of the above, one-class classifiers and generative methods can efficiently model such problems. However, due to the unavailability of data from the abnormal class, training an end-to-end model is a challenging task itself. Therefore, detecting the anomaly classes in unsupervised and semi-supervised settings is a crucial step in such tasks. In this thesis, we propose several methods to model the anomaly detection problem in unsupervised and semi-supervised fashion. The proposed frameworks applied to different related applications of novelty and outlier detection tasks. The results show the superior of our proposed methods in compare to the baselines and existing state-of-the-art methods.

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

The understanding of surroundings and the capability of dynamically and constantly learn the future (unseen) events are ideas that have been in the minds of people during decades [149, 189]. The possibility of detecting the novel situations/distributions of entities/data in a given environment is an essential component for improving the situational understandings of the the system or preventing undesired situations. A dynamic constant learning system should be able to iteratively observe what the consequences of their actions were in the outside world and adapt/learn themselves dynamically according to a given purpose to be accomplished.

The knowledge of the system for being aware of its capabilities and limitation potentially provides the system with the possibility of adapting its decisions in a more appropriated way. This could be done by redefining its own models, learning new models, and selecting different actions for accomplishing a determined task. In this sense, for modeling such dynamic systems as proposed, it is necessary to learn diverse models that relate the different situations of the system. In order to learn diverse models in an unsupervised fashion, the system first needs to detect the deviated situations itself. This process usually referred as Anomaly detection.

Anomaly detection is the process of identifying the new or unexplained set of data to decide whether they are within a previous learned distribution (i.e., inlier in Fig. 1.1) or outside of it (i.e., outliers in Fig. 1.1). The "Anomaly" term is commonly used to refer to the unusual, unseen new observations that do not observed before or is simply different from

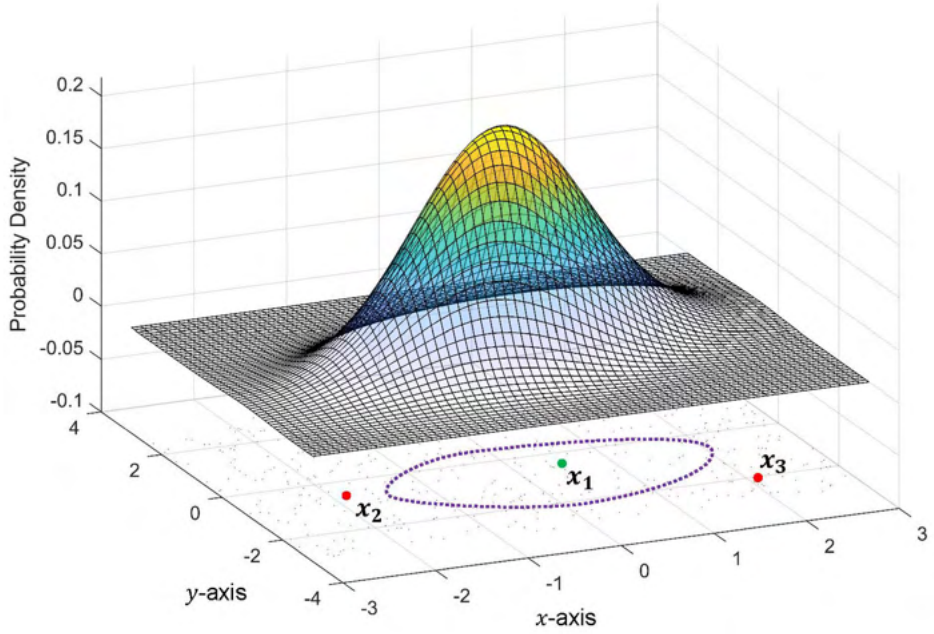


Figure 1.1: Visualization of inlier and outlier samples with respect to the learned distribution by a classifier: As can be seen, x_1 enhanced the inlier sample since it placed inside the classifier decision boundary, where x_2 and x_3 indicates the out-of-distribution samples (outliers).

the other previous observations. Such problems are especially of great interest in computer science studies, as they are very related to outlier detection, novelty detection tasks, and change detection problems. The purpose of this thesis is to introduce learning models based on the essential knowledge of detecting outliers with respect to learned distributions. Such capability can possibly empower the machine learning strategies to increase the awareness of systems.

For that end, in this thesis, we propose approaches for detecting the abnormality of a given data in unsupervised and semi-supervised fashion. We propose to use Autoencoder (AE) and Generative Adversarial Nets (GANs) frameworks, which are trained using normal data in order to learn an internal representation of samples' normality. Since our models are trained with only normal data, they are not able to generate abnormal data. At the testing time, the

real data are compared with the representations reconstructed by our models and abnormal areas are identified by computing local differences between them.

Furthermore, we show the capability of learning more complex situations with the proposed approachs. The proposed self-reinforced representation learning method potentially can be also proved the means of a long-life learning cycle. We propose to create a complementary model that find out the auxiliary information in a given scene in a semantic way, such that a context and a situation, and combine the generated auxiliary information with underlying representations extracted by ordinarily network to extract features. By applying this way of thinking, it has been proposed a methodology to reinforced representation for expressing complex scene situations by using complementary information about given scene/situations. During the time, the system would be able to learn more suitable situation through its adaptive scene understanding capability. The proposed approach not only improves predictions of future events but it can be potentially used for transferring the learned knowledge to other systems. In this context, the situational information related to the environment and contexts perceived by an individual can be moved and interpreted by another body

1.1 Main Contributions

To form a comprehensive analysis, we study different aspects of anomaly detection each evaluated in the aforementioned tasks. For an applicational investigation, we organize this thesis in two parts: *(i)* abnormality detection for visual events, and *(ii)* detecting unusual patterns from images.

- In part *(i)*, we focus on employing the generative models for detecting abnormality of

visual events. Our studies contain applications for the intelligence surveillance system and intelligence vehicle. After an extensive systematic review of existing state-of-the-art approaches, we study the outcomes provided by employing our proposed models for the task of abnormality detection. We first investigate on conventional methods and analyzed the advantages and disadvantages of these approaches. We, next, study utilizing and learning the deep generative models for abnormal event detection. For this purpose, we have introduced various advanced generative models that can improve the discriminativenss of learnt distribution of given data. Additionally, we proposed a self-reinforced representation learning framework to learn condition-adaptive representation by combining auxiliary information and learnt representation of given normal data.

- In part *(ii)*, we explore the significance of utilizing transfer approach between two heterogeneous domains based on unsupervised learning for deriving strict unusual pattern detection method in order to detect the rare pattern in a still image. For this purpose, we first focus on deriving transformation model between two domains, then complement it by applying the stochastic generative approach in order to build a richer representation. We specifically introduce an image-to-frequency transformation model based on GAN to build unsupervised defect detection method for materials. This approach can be learned useful representation without prepared annotation for unusual visual patterns. This sense can be applied into various industrial applications which need to detect unusual visual patterns but there is no prepared annotation for a given data.

The contributions are well described in Chapters 3, 4 and 5, but the main contributions of

this thesis is shortened below with a brief description.

- **Adversarial Prediction Model for Abnormal Event Detection:** We show that the prediction manner which is one of the learning approaches to train deep learning model can be used to effectively derive the generative model and also helpful to detect local anomalies. Specifically, we propose to derive stochastic model by applying the adversarial learning approach for events' past and future. One of the advantages of this approach is that it can derive more strict distribution for normality of given event data. The proposed method is validated on challenging abnormality detection datasets and the results show the superiority of our approach compared with the state-of-the-art methods.

- **Self-reinforced Representation Learning Framework for Driver Drowsiness Detection:** We present a self-reinforced representation learning framework which can provide robust performance to the various real situation without additional components. Particularly, we propose to kernel-level fusion approach that combines semantic information and learnt representations extracted by deep neural networks. One of the advantages of this method is that it can be used not only in the fine-tuning and training phases but also in the testing phase. The proposed method is validated on challenging driver drowsiness dataset and the results show the superiority of our approach compared with the state-of-the-art methods.

- **Adversarial Image-to-Frequency Transformation for Defect Detection:** We introduced a defect detection approach based on cross-domain transformation using Gen-

erative Adversarial Networks (GANs) structured. Each domain is applied to a transformation network to generate the corresponding domain data and used to compute the local distance for detecting the defect (unusual visual patterns) with respect to the corresponding domain in a reconstruction manner. The paradigm of variational autoencoder (VAE) is used to align the distribution of latent representation into the normal distribution. Since this approach does not need prepared annotation for unusual visual patterns, so a model capable to detect abnormal patterns without supervised learning.

1.2 Thesis Overview

Each chapter of this thesis is designed to be self-contained and their contents are structured in such a way that the document as a whole follows the same line and vision. The rest of this thesis is organized as follows.

In Chapter 2, we briefly introduce the preliminaries of generative models. In this chapter, we introduce the background of autoencoder (AE), variational autoencoder (VAE), and generative adversarial network (GAN). We provide theoretic backgrounds and mathematics to apply those approaches for anomaly detection.

In Chapter 3, we first review adversarial prediction models and introduce recurrent adversarial learning for events' past and future. Then, we investigate the possibility of the proposed model as an approach for detecting abnormality of visual events for intelligence surveillance systems. We specifically conduct ablation studies to experimentally demonstrate the efficiencies of both the proposed model and learning approach for identifying anomaly of

events.

In Chapter 4, we describe a self-reinforced representation learning framework which can generate an auxiliary feature and combine it with the representation extracted by CNNs to provide more discriminative power for specific situations. Then, we extend the proposed framework to an application to detect driver drowsiness as a part of advanced driver assistance systems. These methods are based on deep neural networks to discover and learn novel situations.

In Chapter 5, we propose the adversarial cross-domain transformation approach, which is an unsupervised method to detect unusual visual patterns. We apply this method to detect road pavement defect detection. Since this method does not need a prepared sample for unusual patterns, it can be applied to various industrial fields which can provide the large-scale but poorly categorized dataset.

the complexity of distribution and detecting abnormalities with respect to the learned distributions. As a result, these methods are able to model highly diverse and complicated distributions. Such learned models can grant a robust ability for detecting unusual situations in various data distribution to arbitrary abnormal detection methods. This theory is experimentally demonstrated by the experimental results of various applications on several data sets in Chapter 3, 4 and 5.

Chapter 2

Preliminaries: Generative Models

Generative models have been in the forefront of deep unsupervised learning for the last decade. The reason for that is because they offer a very efficient way to analyze and understand unlabeled data. The idea behind generative models is to capture the inner probabilistic distribution that generates a class of data to generate similar data. Generative models have been used in numerous fields and problems such as visual recognition tasks [5], speech recognition and generation [6], and natural language processing [7].

Depending on nature and depth, a model can admit different types of training. In general, some of the training strategies are fast but non-efficient and others are more efficient but hard to carry out or take too long. There are also techniques used to avoid this tradeoff such as two-phased training. The most notable example is deep belief network which often undergoes a separate training for its components (two layers at a time in general) in a phase referred to as pre-training [9], before the final training of the whole network at once in the fine-tuning phase.

2.1 Autoencoder

An autoencoder is a neural network trained for the purpose of recreating its input as the output. It is a feedforward nonrecurrent network of which the aim is to continually reduce the dimensionality to a smaller hidden layer often called the code representative of the input.

In a similar but mirroring process, the network then recreates the same input structure from the code layer. The first part is called the encoder and the second decoder. The goal of an autoencoder is not to perfectly copy the input to the output. Therefore, we must prevent it from learning a trivial identity function which comes easily if the autoencoder is not properly "restrained". The aim is for our model to pick up the underlying patterns and characteristics of the data distribution to be able to generate new never seen before examples of the same distribution as the examples provided during the training phase. Formally, an autoencoder can be written in a deterministic way (although, it is not usually the case) as a composition of two functions:

$$X = f_d(h), where h = f_e(x) \tag{2.1}$$

Where f_e is the encoder, f_d is the decoder, x is the input variable and h is the code.

Since an autoencoder is a particular case of neural networks, it can be trained using the standard techniques for training feedforward neural networks, such as mini batch gradient descent and back-propagation.

2.2 Variational Autoencoder

The strategy in this section can be used to derive a lower bound estimator (a stochastic objective function) for a variety of directed graphical models with continuous latent variables. We will restrict ourselves here to the common case where we have an i.i.d. dataset with latent variables per datapoint, and where we like to perform maximum likelihood (ML) or maximum a posteriori (MAP) inference on the (global) parameters, and variational inference on the

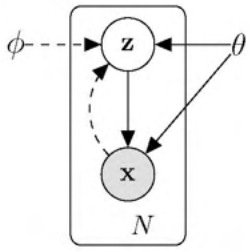


Figure 2.1: The type of directed graphical model under consideration. Solid lines denote the generative model $p_{\boldsymbol{\theta}}(\mathbf{z})p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$, dashed lines denote the variational approximation $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ to the intractable posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$. The variational parameters $\boldsymbol{\phi}$ are learned jointly with the generative model parameters $\boldsymbol{\theta}$.

latent variables. It is, for example, straightforward to extend this scenario to the case where we also perform variational inference on the global parameters; that algorithm is put in the appendix, but experiments with that case are left to future work. Note that our method can be applied to online, non-stationary settings, e.g. streaming data, but here we assume a fixed dataset for simplicity.

Let us consider some dataset $\mathbf{X} = \{\mathbf{x}^{(i)}\}_{i=1}^{N}$ consisting of N i.i.d. samples of some continuous or discrete variable $\mathbf{x}$. We assume that the data are generated by some random process, involving an unobserved continuous random variable $\mathbf{z}$. The process consists of two steps: (1) a value $\mathbf{z}^{(i)}$ is generated from some prior distribution $p_{\boldsymbol{\theta}^*}(\mathbf{z})$; (2) a value $\mathbf{x}^{(i)}$ is generated from some conditional distribution $p_{\boldsymbol{\theta}^*}(\mathbf{x}|\mathbf{z})$. We assume that the prior $p_{\boldsymbol{\theta}^*}(\mathbf{z})$ and likelihood $p_{\boldsymbol{\theta}^*}(\mathbf{x}|\mathbf{z})$ come from parametric families of distributions $p_{\boldsymbol{\theta}}(\mathbf{z})$ and $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$, and that their PDFs are differentiable almost everywhere w.r.t. both $\boldsymbol{\theta}$ and $\mathbf{z}$. Unfortunately, a lot of this process is hidden from our view: the true parameters $\boldsymbol{\theta}^*$ as well as the values of the latent variables $\mathbf{z}^{(i)}$ are unknown to us.

Very importantly, we *do not* make the common simplifying assumptions about the marginal or posterior probabilities. Conversely, we are here interested in a general algorithm that even

works efficiently in the case of:

1. *Intractability*: the case where the integral of the marginal likelihood $p_{\boldsymbol{\theta}}(\mathbf{x}) = \int p_{\boldsymbol{\theta}}(\mathbf{z}) p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})\, d\mathbf{z}$ is intractable (so we cannot evaluate or differentiate the marginal likelihood), where the true posterior density $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}) = p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) p_{\boldsymbol{\theta}}(\mathbf{z}) / p_{\boldsymbol{\theta}}(\mathbf{x})$ is intractable (so the EM algorithm cannot be used), and where the required integrals for any reasonable mean-field VB algorithm are also intractable. These intractabilities are quite common and appear in cases of moderately complicated likelihood functions $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$, e.g. a neural network with a nonlinear hidden layer.

2. *A large dataset*: we have so much data that batch optimization is too costly; we would like to make parameter updates using small minibatches or even single datapoints. Sampling-based solutions, e.g. Monte Carlo EM, would in general be too slow, since it involves a typically expensive sampling loop per datapoint.

We are interested in, and propose a solution to, three related problems in the above scenario:

1. Efficient approximate ML or MAP estimation for the parameters $\boldsymbol{\theta}$. The parameters can be of interest themselves, e.g. if we are analyzing some natural process. They also allow us to mimic the hidden random process and generate artificial data that resembles the real data.

2. Efficient approximate posterior inference of the latent variable $\mathbf{z}$ given an observed value $\mathbf{x}$ for a choice of parameters $\boldsymbol{\theta}$. This is useful for coding or data representation tasks.

3. Efficient approximate marginal inference of the variable $\mathbf{x}$. This allows us to perform all kinds of inference tasks where a prior over $\mathbf{x}$ is required. Common applications in computer vision include image denoising, inpainting and super-resolution.

For the purpose of solving the above problems, let us introduce a recognition model $q_{\phi}(\mathbf{z}|\mathbf{x})$: an approximation to the intractable true posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$. Note that in contrast with the approximate posterior in mean-field variational inference, it is not necessarily factorial and its parameters ϕ are not computed from some closed-form expectation. Instead, we'll introduce a method for learning the recognition model parameters ϕ jointly with the generative model parameters $\boldsymbol{\theta}$.

From a coding theory perspective, the unobserved variables $\mathbf{z}$ have an interpretation as a latent representation or *code*. In this paper we will therefore also refer to the recognition model $q_{\phi}(\mathbf{z}|\mathbf{x})$ as a probabilistic *encoder*, since given a datapoint $\mathbf{x}$ it produces a distribution (e.g. a Gaussian) over the possible values of the code $\mathbf{z}$ from which the datapoint $\mathbf{x}$ could have been generated. In a similar vein we will refer to $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ as a probabilistic *decoder*, since given a code $\mathbf{z}$ it produces a distribution over the possible corresponding values of $\mathbf{x}$.

The marginal likelihood is composed of a sum over the marginal likelihoods of individual datapoints $\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(N)}) = \sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)})$, which can each be rewritten as:

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) = D_{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}^{(i)})||p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^{(i)})) + \mathcal{L}(\boldsymbol{\theta}, \phi; \mathbf{x}^{(i)}) \tag{2.2}$$

The first RHS term is the KL divergence of the approximate from the true posterior. Since this KL-divergence is non-negative, the second RHS term $\mathcal{L}(\boldsymbol{\theta}, \phi; \mathbf{x}^{(i)})$ is called the (variational)

lower bound on the marginal likelihood of datapoint i, and can be written as:

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) \geq \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\left[-\log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) + \log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})\right] \tag{2.3}$$

which can also be written as:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = -D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})||p_{\boldsymbol{\theta}}(\mathbf{z})) + \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\mathbf{z})\right] \tag{2.4}$$

We want to differentiate and optimize the lower bound $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)})$ w.r.t. both the variational parameters $\boldsymbol{\phi}$ and generative parameters $\boldsymbol{\theta}$. However, the gradient of the lower bound w.r.t. $\boldsymbol{\phi}$ is a bit problematic. The usual (naïve) Monte Carlo gradient estimator for this type of problem is: $\nabla_{\boldsymbol{\phi}}\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z})}\left[f(\mathbf{z})\right] = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z})}\left[f(\mathbf{z})\nabla_{q_{\boldsymbol{\phi}}(\mathbf{z})}\log q_{\boldsymbol{\phi}}(\mathbf{z})\right] \simeq \frac{1}{L}\sum_{l=1}^{L} f(\mathbf{z})\nabla_{q_{\boldsymbol{\phi}}(\mathbf{z}^{(l)})}\log q_{\boldsymbol{\phi}}(\mathbf{z}^{(l)})$ where $\mathbf{z}^{(l)} \sim q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})$. This gradient estimator exhibits exhibits very high variance and is impractical for our purposes.

2.3 Generative Adversarial Network

Generative adversarial networks (GANs) [8] have shown impressive achievements in many computer vision tasks. GANs employ a two-player game theory using a minimax strategy, where two different networks are trained concurrently in an unsupervised manner. That is, a generator $\mathcal{G}$ tries to produce realistic samples, while a discriminator $\mathcal{D}$ is trained to classify the real sample from the fake sample generated by the generator. The minimax objective

function can be defined as follows:

$$\min_{\theta_{\mathcal{G}}} \max_{\theta_{\mathcal{D}}} V(\mathcal{D}, \mathcal{G}) = \mathbb{E}_{x \sim p_{real}\,(x)}[log\mathcal{D}(x)] + \mathbb{E}_{z \sim p_{fake}(z)}[log(1 - \mathcal{D}(\mathcal{G}(z)))] \tag{2.5}$$

where z is the random noise and x denotes the real sample, $\theta_{\mathcal{G}}$ and $\theta_{\mathcal{D}}$ denote the parameters of $\mathcal{G}$ and $\mathcal{D}$ respectively.

Finding an optimal distribution of $\mathcal{D}$ with $\mathcal{G}$ fixed is equivalent to minimizing the Jenson-Shannon (JS) divergence of original GAN. A lot of recent video prediction and generation work with GANs have focused on unsupervised settings [9] or semi-supervised settings with stochastic conditional methods [10]. Simply optimizing the GAN objective function utilizes a learned loss function through the $\mathcal{D}$, which can produce adequate predictions. However, these works without a regularization technique [9, 10] are prone to show inconsistent training process such as vanishing gradient on the generator $\mathcal{G}$ and even the model collapse issue. These issues are concerned as the potential risks in making a reliable video prediction or generation models. To address these issues, Arjovsky *et al.*,[11] suggested using the Wasserstein distance, which has a much smoother value space than JS divergence, in computing the difference between the real sample and produced samples.

While promising results have been achieved by using better theoretical characteristics, it remains a challenge in terms of the optimization process, *i.e.* vanishing or exploding gradients, primarily due to the use of weight clipping to enforce a 1-Lipschitz constraint on the $\mathcal{D}$. Thus, an improved version of WGAN [12] with a gradient penalty is proposed to enforce the Lipschitz constraint, accelerating the convergence. Based on the above approach for better

gradient behaviour, the GAN network can be solved by the following minimax problem:

$$\begin{aligned}\min_{\theta_{\mathcal{G}}} \max_{\theta_{\mathcal{D}}} V(\mathcal{D}, \mathcal{G}) = &- \mathbb{E}_x[\mathcal{G}(x)] + \mathbb{E}_z[\mathcal{D}(\mathcal{G}(z))] \\ &+ \lambda \mathbb{E}_{\hat{x}}(\|\nabla_{\hat{x}} \mathcal{D}(\hat{x})\|_2 - 1)^2],\end{aligned} \tag{2.6}$$

where the first two terms denote original critic loss that calculates a Wasserstein distance estimation, and the last term denotes gradient penalty as regularizer of the network; the point $\hat{x}$ is uniformly sampled between the data distribution and generated sample distribution; λ is defined as a constant weight parameter. Specifically, this formulation removes the log function in the minimax losses compared to the original GAN. In this paper, we employ WGAN for event prediction under the consecutive frames. To the best of our knowledge, this is the first time to apply WGAN to the event prediction model to handle the abnormal event detection.

Chapter 3

Abnormal Event Detection for Intelligence Surveillance System

Detection of abnormal events (a.k.a. unusual events, anomalous behaviours,and rare events) is one of the challenging issues in computer vision research, and it is applicable to various surveillance applications such as accident detection on the road, intrusion detection, and criminal behaviour detection. Previously, there is no clear definition of abnormal event universally accepted. Cong et al. [13] define two types of abnormal events in videos: 1) Local abnormal event (LAE) and 2) Global abnormal event (GAE). LAE is defined as the behaviour of an individual that is different from behaviours of spatially adjacent neighbours. GAE is identified as group behaviours in global scenes representing unusual patterns. In evaluation tasks of studies for the abnormal event detection, GAEs and LAEs are evaluated as results of the frame-level abnormal event detection and the pixel-level abnormal event detection respectively.

3.1 Abnormal Event Detection

In general, abnormal events can be identified as irregular events from normal ones. The scheme to identify anomalous events is basically to classify abnormal events using a given training dataset which consists of normal event samples. Conventional approaches treat the

abnormal event detection as one-class classification, outlier detection, or anomalous pattern detection, and they consider an event as an abnormal event if it has a lower likelihood of belonging to a model fitted over training samples containing normal events only. These philosophies for the abnormal event detection gave direction to developing a good event detection method: One is the representation of behaviours (Event representation) and the other is the construction of a model for normal events (Normal model construction).

In the studies for event representation, researchers have focused on finding hand-crafted features that describe rich and discriminative information that can help to distinguish between the normal and abnormal events. In early studies [14, 15, 16] for event representation, researchers used tracking methods [17] to extract the trace information about moving objects, and they classified irregular tracking patterns as the abnormal. Some researchers [18, 19, 20, 21, 22] tried to model the movements of crowds to overcome shortcoming mentioned above, such as social force model (SFM) [20], statistics or patterns of optical flow in local regions [18, 21], the tracklets which are short-term tracking results of interesting points [22], histogram of gradient (HoG) and the histogram of optical flow (HoF) [23], and space-time gradient based scene representation methods [24, 25, 26].

The research on the construction of normal event models gives regularized computation methods for abnormality of events that are robust to image transformation group such as rotation, translation, and blurring. To model a normal event, most conventional approaches intend to identify an input data with lower likelihood as an abnormal by fitting a probabilistic model using a training dataset. There are various approaches, such as Markov random field (MRF) [21], Bayesian networks [25, 27, 27], Hidden Markov Model (HMM) [25], Latent

Dirichlet Allocation (LDA) [20],and Multi-scale Motion Interrelated Patterns (SMMIP or MIP) [28]. A few years ago, the sparse representation began to be used to model previous normal behaviours with a dictionary learning approach [13, 26].

Recently, deep learning architectures have been successfully used to solve various computer vision problems, such as image recognition [29], object detection [30], gesture recognition [31], and action recognition [32]. In particular, 3D convolutional neural network shows remarkable performance in analyzing the video stream for behaviour understanding [33, 34]. In abnormal event detection, Based on the deep learning, various methods such as three-channel auto-encoder [35], spatio-temporal auto-encoder [36], and recurrent network using the long short-term memory (LSTM) [37], have been proposed to detect an anomalous of events. Although various methods [38, 39, 40] to extract superior hand-crafted features have been proposed, these methods cannot provide sufficient representation of the countless motion pattern observed in nature. The key to success is that using deep learning architectures, rich and discriminative features can be learned via cascaded non-linear transformations automatically. Thus, it is reasonable to expect that classifying rare events in a video can also benefit from deep learning models.

3.2 Preview Works

There exists a large number of approaches for for abnormality detection in video sequences. In the following some of leading approaches are described. The major challenge in abnormality detection is that there is not a clear definition of abnormality, since they are basically context dependent and can be defined as outliers of normal distributions. Based on this widely accepted

definition of abnormality, existing approaches for detecting abnormal events in crowds can be generally classified into two main categories: *i)* object-based approaches, and *ii)* holistic techniques.

3.2.1 Object-based approaches

Object-based methods treat a crowd as a set of different objects. The extracted objects are then tracked through the video sequence and the target behavioral patterns are inferred from their motion/interaction models [41] (e.g. based on trajectories [42]). This class of methods relies on the detection and tracking of people and objects [43]. Despite promising improvements to address several crowd problems [44, 45, 46, 47], they are limited to low density situations and perform well when high quality videos are provided, which is not the case in real-world situations. In other words, they are not capable of handling high density crowd scenarios due to severe occlusion and clutter which make individuals/objects detecting and tracking intractable [48, 49]. Some works made noticeable efforts to circumvent robustness issues. For instance, Zhao and Nevatia [50] used 3D human models to detect persons in the observed scene as well as a probabilistic framework for tracking extracted features from the persons. In contrast, some other methods track feature points in the scene using the well-known KLT algorithm [45, 51]. Then, trajectories are clustered using space proximity. Such a clustering step helps to obtain a one-to-one association between individuals and trajectory clusters, which is quite a strong assumption seldom verified in a crowded scenario.

3.2.2 Holistic approaches

The holistic approaches, in contract with Object-based approaches, do not aim to separately detect and track each individual/object in a scene. Instead, they treat a crowd as a single entity and try to employ low/medium level visual features extracted from video frames to analyze the crowd scene as a whole [52, 53, 54]. The differences among holistic approaches are regarding the way that they represent the scene as well as the way that they detect anomaly. Some of them use statistic (machine learning) techniques in order to learn the "normal" behavior of a crowd in a given environment and define abnormality as those situations having a low probability value under the so constructed probabilistic framework [55, 56]. Differently, abnormality can be defined using model based techniques (i.e., methods not involving statistic estimations), for instance dealing with abnormality detection as a saliency detection problem or using ad-hoc rules for finding specific patterns in the optical-flow data. Below the holistic approaches are classified with respect to the way the scene is represented and not the way in which the anomaly is detected.

Optical-flow histogram analysis. The simplest way to represent the global movement in a crowded scene is probably using simple statistics extracted from the optical flow data [57]. Zhong *et. al.* [58] propose an unsupervised method which use the "hard to describe" but "easy to verify" property of unusual events without any explicit modeling of the normality. They proposed to use only simple motion models without supervised feature selections, however the method may fail in the crowded scenario and more complex activity patterns. Krausz and Bauckhage in [52, 59] used the histogram of the optical flow as the basic representation. Simple heuristic rules are then proposed in order to detect specific crowd dangerous behaviors,

such as congestion or turbulence situations. For instance, a congestion situation is detected by looking for symmetric pairs of histograms extracted from consecutive frames, which indicate slow lateral oscillations of the people's upper body.

Spatio-temporal grids. Different approaches deal with the complexity of a dynamic scene analysis by partitioning a given video in spatio-temporal volumes [60, 53]. Each frame is partitioned in a spatial grid of $n \times m$ cells, and the frame sequence is partitioned in k consecutive frames, which altogether brings to a $n \times m \times k$ spatio-temporal volume. In [25, 61] Kratz and Nishino extract spatio-temporal gradients from each pixel of a given frame. Then, the gradients of a spatio-temporal cell are modeled using Spatio-Temporal Motion Pattern Models, which are basically 3D Gaussian clusters of gradients. A simple leader follower on-line clustering algorithm is used to group gradients observed at training time in separate cluster centers (prototypes). At testing time, a single Gaussian cluster is extracted from each cell of the input video and the Kullback-Leibler distance is used in order to select the training prototype with the closest gradient distribution. Finally, a mixed spatio-temporal Hidden Markov Model is used in order to model transition probabilities among prototypes.

Mahadevan et al. [62] model the observed movement in each spatio-temporal cell using dynamic textures, which can be seen as an extension of PCA-based representations. Whereas PCA spaces only model the appearance of a given patch texture, dynamic textures also represent the statistically valid transitions among textures in a patch. In each cell, all the possible dynamic textures are represented with a Mixture of Dynamic Textures model, which gives the probability of a test patch to be anomalous. In this way, the authors show that not only temporal anomalies but also pure appearance anomalies can be detected. In the same work the

authors present also an interesting definition of spatial saliency based on Mutual Information between features and foreground/background classes. In the proposal of Mahadevan et al., only local (cell-level) abnormalities are detected.

Physics inspired models. Some research groups exploit mathematical models derived from fluid dynamics or other physics laws in order to model a crowd as an ensemble of moving particles. The Social Force model to describe the behavior of a crowd as the result of interaction of individuals uses the Second Newton's law to describe the causes of the movement of a set of particles [63]. In [20] the Social Force model is used to detect anomalies and estimate local anomalies by detecting regions in the current frame in which the local optical flow is largely different from the average optical flow computed in the neighboring regions. Randomly selected spatio-temporal volumes of Force Flow are used to model the normal behavior of the crowd and classify frames as normal and abnormal cases using a bag of words approach. The same research group uses Coherent Structures and a fluid dynamics framework to segment optical flow data in dynamically coherent clusters [64]. Anomalies are detected looking at sharp differences between the segmentation outcomes of consecutive frames. Finally, the Shah's group in [65] proposes a method to classify the critical points of a continuous dynamical system. They represent the scene as a grid of particles initializing a dynamical system which is defined by the optical flow information. Such simplification provides a linear approximation of the input complex dynamical system, aiming to identify typical crowd behaviors. In [66] a novel method is presented for detecting and localizing anomalies in complicated crowd sequences using a Lagrangian particle dynamics approach, together with chaotic modeling. All these works do not detect and track individuals. Instead,

they apply particle advection technique that places particles onto a grid and moves them according to the underlying optical flow field. However, particle advection is not applicable when the camera viewpoints are inappropriate and the resulted occlusions in situations of high pedestrian density.

Segmentation approach. In scenarios with high density crowds, i.e. political rallies, [65] religious festivals and marathons which involve, the large gatherings of people poses significant challenges from the scene monitoring point of view, where current automated surveillance systems fail to deal with such cases. The reason for such failure is the difficulty of detection and tracking of target objects in high density scenes. In such cases, segmenting high density scenes into dynamically and physically meaningful flow segments can be a tractable solution. Such emerging motion patterns are called as "flow segment". Using instantaneous motions of a video, i.e. The motion flow field is another viable solution presented in [67]. The motion flow field is a union of independent flow vectors computed in different frames and a set of flow vectors representing the instantaneous motions in a video. They first use existing optical flow method to compute flow vectors in each frame and then combine them into a global motion field. This flow field may contain thousands of flow vectors, and, therefore, it is computational expensive to obtain the shortest paths based on such a large number of data. Detecting motion patterns in this flow field can therefore be formulated as a clustering problem of the motion flow fields. A hierarchical agglomerative clustering algorithm is applied to group flow vectors into desired motion patterns.

Tracklet-based approach. Tracklets are typically short temporal sequences of points, commonly extracted using the KLT method [51]. The points to track can be salient points,

randomly selected points or densely distributed points on a grid [54]. Tracklet-based methods can be seen as a trade-off between object and holistic based approaches. In fact, on the one hand they model the observed scene using trajectory analysis but on the other hand they do not rely on person detection and tracking, being focused on tracking of simple points which is a much simpler task. Zhou et al. [68] dealt with a tracklet as a document in a Bag of Word approach. All the points of a given tracklet are associated with words of a codebook, according to their location and velocity directions. Then, tracklets are automatically clustered in "topics", where each topic describes a semantic region in the scene. Temporal and spatial dependencies between tracklets are modeled using two different MRFs, one for the spatial and one for the temporal dependence respectively. In [69] the same authors go a step forward and use tracklet clusters in order to classify a test tracklet according to the closest cluster. In the same work, past and future paths of the individual represented by the analyzed tracklet can be simulated.

3.2.3 Deep learning-based approaches

There is a wealth of literature on abnormality detection that is based on hand-crafted features (e.g., Optical-Flow, Tracklets, etc.) to model the normal activity patterns, whereas deep learning-based approaches yet is not studied well. Despite the recent improvements in deep learning on different areas including image classification [70, 71, 72], object and scene detection/recognition [73, 74], image segmentation [75, 76], human action recognition [77, 78] vision and language integration [79, 80, 81], and human-machine collaboration [82, 83, 84], but still the deep learning approaches are not exploited well for abnormality

detection task [85, 86, 87, 88, 89]. This is mainly due to the nature of of abnormality detection task and the lack of annotated data. Deep networks are data hungry and training a network with minimal data is a challenging task itself [90, 91, 92]. Hence, the deep learning-based works for abnormality detection task, mainly use existing Convolutional Neural Network (CNN) models trained for other tasks (e.g., object recognition) which are adapted to the abnormality detection task. For instance, Ravanbakhsh et al. [87] proposed a Binary Quantization Layer, plugged as a final layer on top of a pre-trained CNN, in order to represent patch-based temporal motion patterns. However, the network proposed in [87] is not trained end-to-end and is based on a complex post-processing stage and on a pre-computed codebook of the convolutional feature values. Similarly, in [88, 89], a fully convolutional neural network is proposed which is a combination of a pre-trained CNN (i.e., AlexNet [70]) and a new convolutional layer where kernels have been trained from scratch. Sabokrou et al. [93] introduce a patch-based anomaly detection framework based on an Autoencoder (AE) reconstruction error and sparsity constraints. However this work is limited to a single modality setup. Stacked Denoising Autoencoders (SDAs) are used by Xu et al. [35] to learn motion and appearance feature representations. The networks used in this work are relatively shallow, since training deep SDAs on small abnormality datasets can be prone to over-fitting issues and the networks' input is limited to a small image patch.

Moreover, after the SDAs-based features have been learned, multiple one-class SVMs need to be trained on top of these features in order to create the final classifiers, and the learned features may be sub-optimal because they are not jointly optimized with respect to the final abnormality discrimination task. Feng et al. [94] use 3D gradients and a PCANet [95] in

order to extract patch-based appearance features whose normal distribution is then modeled using a deep Gaussian Mixture Model network (deep GMM [96]). Also in this case the feature extraction process and the normal event modeling are obtained using two separate stages (corresponding to two different networks) and the lack of an end-to-end training which jointly optimizes both these stages can likely produce sub-optimal representations. Furthermore, the number of Gaussian components in each layer of the deep GMM is a critical hyperparameter which needs to be set using supervised validation data.

The only deep learning based approach proposing a framework which can be fully-trained in an end-to-end fashion we are aware of is the Convolutional AE network proposed in [97], where a deep representation is learned by minimizing the AE-based frame reconstruction. At testing time, an anomaly is detected computing the difference between the AE-based frame reconstruction and the real test frame.

3.3 Abnormal Event Detection Dataset

In recent years, the number of studies on abnormal behaviour detection has grown rapidly in both academic and commercial fields. This comes with a grown demand for public datasets to use for video surveillance system evaluation, yet there are not many available public datasets for abnormal event detection.

In our experiments we used several publicly available datasets for evaluation, including UCSD [62], UMN [20], and UCF-Crime [101] datasets. Examples of video frame scenes is shown in Fig. 3.1. A list of datasets is presented in Tab. 3.1. Example frames of normal and abnormal events of each dataset are represented in Fig 3.1.

Dataset	Sequences	Description
UCSD [62]	two subsets: *PED1*: 70 videos (34 training, 36 testing), 158×238 pixels. *PED2*: 30 videos (16 training, 14 testing), 240×360 pixels.	normal events define as pedestrians in the walkways, and non-pedestrians correspond to abnormal events.
UMN [20]	including 11 videos from three different indoor and outdoor scenes. Resolution of 240×320 pixels.	each sequence starts with a normal scene and ends with crowd dispersion as an abnormal event.
Subway [18]	two videos called *Entrance gate* and *Exit gate* with 512×384 pixel resolution. *Entrance gate* has 1 hour 36 min run time that consists of 144,249 frames. *Exit gate* has 43 min of run time which is composed of 64,900 frames.	videos are captured from various public cameras installed in a subway station.
Avenue [98]	16 training videos containing 15,328 frames and 21 test videos with 15,324 frames. The resolution of each video is 360×640.	The dataset includes several challenging issues such as camera shaking, outliers in training videos, and absence of motions on the part of training videos.
UCF Crime [99]	1900 videos with various resolutions.	videos are captured from various surveillance cameras, and consists of long untrimmed surveillance videos.

Table 3.1: Event anomaly detection datasets [100].

UCSD dataset. consists of two datasets captured from different scenes: PED1 and PED2.

- PED1 contains 34/16 training/test sequences with frame resolution 238×158 pixels. Video sequences indicate groups of individuals walking toward and away from the camera, with a certain degree of perspective distortion. The dataset consists of 3,400 abnormal frame samples and 5,500 normal frames.

- PED2 includes 16/12 training/test video samples, with about 1,600 abnormal frames and 350 normal samples. This subset indicates a scene where most pedestrians move

Figure 3.1: From the left, each column shows sample frames for normal and abnormal events of UMN, UCSD, Avenue, Subway, and UCF-Crime datasets. Yellow boxes denote the locations of abnormal events of each abnormal frame.

horizontally. The frames resolution is 360×240.

This dataset is challenging due to different camera view points, low resolution, different types of moving objects across the scene, presence of one or more anomalies in the frames. The video footage of each scene is divided into two subsets: test and training (only normal conditions).

UMN dataset. contains 11 different scenarios in three different indoor and outdoor situations. UMN includes 7700 frames in total with frame resolution of 320×240 pixels. All sequences start with a normal scene and end with abnormality section.

Avenue dataset [98] consists of 16 training videos containing 15,328 frames and 21 test videos with 15,324 frames. The resolution of each video is 360×640. The dataset includes several challenging issues such as camera shaking, outliers in training videos, and absence of motions on the part of training videos. This dataset provides pixel-level annotation for AED.

Subway dataset [18] includes two videos called *Entrance gate* and *Exit gate* with 512×384 pixel resolution. *Entrance gate* has 1 hour 36 min run time that consists of 144,249 frames. *Exit gate* has 43 min of run time which is composed of 64,900 frames. The abnormal events on this dataset are defined by motion of pedestrian in the wrong direction, *e.g.* jumping motion of pedestrian for avoiding payment of subway fee. Subway dataset only provides an event-level ground-truth so that detection results are transformed into the event-level results [18].

UCF-Crime dataset [99] consists of long untrimmed surveillance videos which cover 13 real-world anomalies, including Abuse, Arrest, Arson, Assault, Road Accident, Burglary, Explosion, Fighting, Robbery, Shooting, Stealing, Shoplifting, and Vandalism. These anomalies are selected because they have a significant impact on public safety.

3.3.1 Evaluation metrics

We considered two different types of experiments for an abnormal events detection: 1) Global abnormal events (GAEs): frame-level abnormal event detection, and 2) Local abnormal events (LAEs): pixel-level abnormal event detection. To evaluate the detection performance for GAEs, initially, we compute true positive rate (TPR) which is the ratio between the number of true positive frames (truly labelled frame as an abnormal) and the number of positive frame (identified frame as an abnormal), and false positive rate (FPR) calculated as the ratio between the number of false positive frames (truly labelled frame as a normal) and the number of

negative frames (identified frame as a normal). TRP and FPR are defined as follows:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \quad \mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}},$$

where TP and FN indicate true positive and false negative. FP and TN denote false positive and true negative. All components are computed by matching the detection results and the given ground-truth.

For the evaluation of pixel-level detection performance, we define the correctness of pixel-level abnormal event detection as the overlapped scale between the localized area deemed unusual and the ground-truth. When it is larger than 40%, the input is classified as the abnormal [20].

Among the several metrics to evaluate a video surveillance system, two of them are commonly used, and both criteria are computed from the Receiver Operating Characteristic (ROC) Curve:

- Area Under Roc curve (AUC): The AUC mainly is used for performance comparison in different tasks [100].
- Equal Error Rate (EER): The EER is the point on a ROC curve where the false positive rate (i.e., normal detects as abnormal) is equal to the false negative rate (i.e., abnormal is classified as normal).

The performances of a system can be considered as good if the value of the AUC be as high, while the value of the EER is as small as possible. In our works, we followed these standard evaluation metrics and present our results in terms of ROC curves, AUC and ERR.

3.4 Joint Learning of Motion and Appearance

Abnormal events in videos are observed with a various visual patterns related to a change of an appearance and a motion included in the videos. In this section, we describe the proposed framework including the joint learning method to discover the informative representation from the pure appearance and motion information simultaneously and the end-to-end learning framework for an abnormal event detection.

3.4.1 Joint learning of motion and appearance

3D-DCNN [102] is usually designed with a single input structure in most computer vision tasks, such as image recognition [103] and action recognition [102]. One input and a corresponding output of the 3D-DCNN in these studies are a sample and a predicted label respectively. However, a network architecture with a single input channel is not suitable for detecting the various types of abnormal events, since anomalous patterns of events in videos are accompanied by various changes in motion and appearance, and sometimes these changes could not be distinguished dichotomically. To deal with this problem, we introduce a learning method for a joint spatio-temporal representation for appearance and motion using the 3D-DCNN which is based on a dual input structure. This structure is designed for extraction of a joint feature from two independent inputs.

First, we give the notations in our work. An input data $v = \{v_a, v_m\}$ consists of an appearance 3D volume $v_a \in R^{w \times h \times c \times t}$ and a motion 3D volume $v_m = R^{w \times h \times c \times t}$, where w and h denote the width and height of each 3D volume, c and t are the number of channels and temporal length of 3D volume. In this work, we assume that the dimension of the appearance

and motion 3D volumes are equal except the number of channels c. In recent research on the abnormal event detection, various methods were offered for extracting discriminative features which include ther interaction force of the social force model [20], 3D histogram of gradient (3DHoG) [26], and social-aware attribute [104], which are more complex hand-crafted features. While the proposed method is agnostic to the particular feature extraction method, we use simply normalised image and pure dense optical flow as inputs of appearance and motion to enable a controlled comparison with previous works. The framework is composed of two models: a representation model f_{re} and a model for an abnormal event detection f_{de}, and each model is defined by the parameter $\theta = \{W, b\}$, where W is a weight and b is a bias. For any input sample, $\hat{o}$ denotes the output of the model and o denotes ground-truth of the input sample. The representation model f_{re} needs to learn the joint spatio-temporal representation from each input data $v = \{v_a, v_m\}$. We apply a joint learning scheme instead of learning two separate representations. Given an input data v that represents the joint feature, we can discover the appearance and motion information simultaneously to distinguish abnormal events from normals. The representation model to extract a joint representation is given by

$$\boldsymbol{\alpha} = f_{re}(v; \theta_{re}), \quad \boldsymbol{\alpha} \in R^{w_\alpha \times h_\alpha \times c_\alpha \times t_\alpha} \tag{3.1}$$

where θ_{re} is the set of parameter containing the set of weights W_{re} and the set of biases b_{re}, which are dependent on the depth of the network structure. To combine the two different vectors and subsequently learn the joint representation, we apply the early fusion method

[105] to learn the deep joint representation. The early fusion is given by

$$\alpha^{xyt} = \sigma[\sum_{p}^{w_e}\sum_{q}^{h_e}\sum_{k}^{t_e}(v_a^{pqk}W_a{}^{pqk} + v_m^{pqk}W_m{}^{pqk}) + b_e] \tag{3.2}$$

where σ is an activation function in the joint representation learning model, and we employed the rectified linear units [106] in this work. α^{xyt} is the initial joint representation of the unit at position (x, y, t) in the early fusion layer. v_a and v_m are the input 3D volume corresponding to the output α^{xyt} geometrically. W_a, W_m, and b_e are the weights of appearance, motion inputs and the bias for the early fusion layer. The weights denote the kernel (3D local receptive field) of a 3D convolutional neural network.

The early fusion is essential for learning a robust joint representation because the early fusion allows the network to precisely learn partial appearance and motion [105]. In the context of representation learning, we expect latent features to represent basic patterns that can be optimized with the sparse property. Hence we further append the initial fusion to the learning of joint representation. The architecture of the representation model contains a total of eight convolutional layers: one joint 3D convolutional layer, seven general 3D convolutional layers. The architectural detail of the representation model is illustrated in figure 3.2.

3.4.2 Abnormality detection with joint learning

The event representation model described in the previous subsection generates a set of combined features, which provides a joint description containing information on the appearance and motion of an input data $v \in \{v_a, v_m\}$. The feature $\boldsymbol{\alpha}$ in Eq. (3) obtained from the representation model is directly applied to the detection model f_{de} to compute

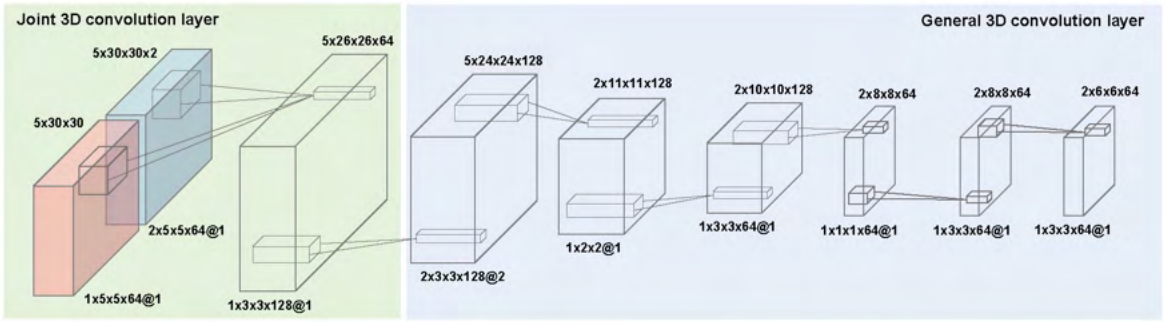


Figure 3.2: The illustration of the architecture detail of the joint representation model based on 3D-DCNN. The red and blue boxes are 3D volumes of appearance and motion. An initial convolution layer is the 3D joint convolution layer, and the layers behind the initial layer are general convolution layers. Numbers that are located above each layer denote the structural information of each layer, and numbers below the layers represent the dimensional structure of the local receptive field and the scale of padding.

the likelihood of input samples. The distribution of the joint representations containing the abstracted information of input appearance and motion volumes is more likely non-linear since there is huge variation in the patterns of normal and abnormal events. Thus, we employ the additional fully connected network for accurately classifying the joint representations distributed non-linearly, although it raises the computational complexity by increasing the number of parameters. Our abnormal event detection model is a three-layer neural network with two fully connected networks with 256 units and one softmax layer with two units. This fully connected deep neural network which will be on top of the joint representation learning model is formulated as:

$$\hat{o} = f_{de}(\boldsymbol{\alpha}; \theta_{de}), \quad \hat{o} \in R^{2\times 1} \tag{3.3}$$

where $\hat{o}$ denotes the output of detection model, and θ_{de} is the set of model parameters consisting of the weights $W = \{W^1_{de}, W^2_{de}, W^2_{de}\}$ and the biases $b = \{b^1_{de}, b^2_{de}, b^3_{de}\}$. The output consists of two units for describing the normality and abnormality of the result of the

soft-max function. The two fully connected networks are represented as follow:

$$\beta_1 = \sigma(\boldsymbol{\alpha} W_{de}^1 + b_{de}^1), \tag{3.4}$$

$$\beta_2 = \sigma(\beta_1 W_{de}^2 + b_{de}^2), \tag{3.5}$$

where W_{de}^1 and W_{de}^2 are the weight parameters of the first and second fully connected layers respectively, and b_{de}^1 and b_{de}^2 denote the biases parameters of the layers. β_1 and β_1 are the output of the first and second layers of the fully connected network. In the final layer, we attempt to compute a likelihood based on the soft-max function, represented as:

$$\hat{o} = softmax(\beta_2 W_{de}^3 + b_{de}^3) \tag{3.6}$$

where $\hat{o}$ is the output of the soft-max layer, and W_{de}^3 and b_{de}^3 are the weight and bias parameters for the layer. Each value of the soft-max layer of the event detection model reflects the normal and abnormal degrees of input samples. Using the output of the soft-max layer in Eq. (6), we can detect an anomalous event in each input 3D volume separately. A high value of the normal unit signifies that an input sample is likely to the normal events, and therefore the high value of an abnormal unit signified that it is be abnormal. The dimensionality of each hidden layer is 256, and the outputs of each hidden layer are calculated by the rectified linear unit [106] identically to the joint representation learning. Only final layer takes their output using the softmax function for estimation of the probability of abnormal events of each input.

For event detection from the input appearance and motion 3D volumes, first using the

joint representation model, we extract the joint spatio-temporal features and then determine whether the input 3D volumes contain abnormal event pattern or not. In our scheme, the value of the output layer of the proposed framework is calculated by the soft-max function $\frac{e^{x_i}}{\sum_{k=1}^{2} e^{x_k}}$ which can represent the likelihood for normal and abnormal events simultaneously. However, simply comparing the two values of the output layer can generate a lot of false positives, since a tiny difference in the two values can also be classified as abnormal or normal. To control the sensitivity of the model for detecting the abnormal events, we apply a specific threshold such that, if the value of the abnormal unit is greater than the threshold, the input sample will be classified as an abnormal one.

The proposed framework presents the location of an anomalous event using the index of a volume since it provides the volume-level event detection. An input is divided into fixed size volumes and each volume has an index corresponding to their relative position in the frame. By using this index, the proposed framework can represent the location of an anomalous event within a frame.

3.4.3 Training and inference

Before training the entire framework, we pre-train the representation model using an unsupervised learning approach based on stacked convolutional autoencoder proposed by Masci et al. [107]. We refer to the network structure in [107] to built the convolutional autoencoder, which contains dual inputs. It allows us to avoid a poorly and locally optimized solutions of the representation model. We employ the learning method of the stacked convolutional autoencoder for pre-training of the proposed dual-input 3D Deep convolutional neural network.

The dual-input convolutional autoencoder is intuitively similar to the stacked convolutional autoencoder. For two different input vectors x and y, the latent representation of the early fusion feature map and the latent representation of followed feature map are given by

$$h^0 = \sigma(x * W_x^0 + y * W_y^0 + b_e^0), \tag{3.7}$$

$$h^k = \sigma(h^{k-1} * W^k + b^k), \tag{3.8}$$

where W_x^0 and W_y^0 are the weight matrices of the early fusion layer, and b_e^0 is the bias of the early fusion layer in the encoding part. W^k and b^k are the weight matrix and the bias of the k^{th} feature map. By using the convolutional network structure, the two inputs are encoded to an abstracted dimensional vector, and the encoded results are reconstructed by a reverse mapping based on the learnt weight matrix in the encoding part. However, the reconstruction function is separated in the final decoding layer since the proposed 3D-DCNN consists of two input vectors, represented as

$$\hat{x} = \sigma(\hat{h} * W_x'^0 + b_d^0), \tag{3.9}$$

$$\hat{y} = \sigma(\hat{h} * W_y'^0 + b_d^0), \tag{3.10}$$

where $\hat{x}$ and $\hat{y}$ are the reconstructed results of the two input vectors, and $W_x'^0$ and $W_y'^0$ are the transposed weight matrices of the two input vectors in the early fusion layer. $\hat{h}$ is the reconstructed latent representation of the adjacent convolutional layer, and b_d^0 is the bias of the decoding part. Consequently, the optimization problem for pre-training of the representation

model is expressed as follows:

$$argmin_{W,b_e,b_d} \Sigma_i^N E((x_i, y_i), (\hat{x}_i, \hat{y}_i)) \tag{3.11}$$

In our proposed model, the two input vectors are denoted as the 3D volumes of appearance v_a and motion v_m, and the output vectors are defined as the reconstruction vectors associated with the input vectors, and N is the number of training samples. We use the back-propagation algorithm [108] and $l2$-loss function for the pre-training. We employed batch training with 32 size for 100 epochs. The initial learning rate is set to 0.1 and is decayed by multiplying the 0.1 at every 10 epoch, the final learning rate is fixed at 0.0001.

We train the entire framework after pre-training of the event representation model. An optimization scheme for the entire framework operates under the detection objective. Our abnormal event detection model is trained by minimizing the detection loss using the ground truth for normal and abnormal events, described as follows:

$$\min_{\theta_{re},\theta_{de}} \Sigma_i^N E(o, \hat{o}), \tag{3.12}$$

where o and $\hat{o}$ are an annotation value and the output of network in Eq. (6) associated with the input sample $v \in \{v_a, v_m\}$, and E denotes the cost function for the overall architecture containing the representation model and the detection model. We employ the cross-entropy loss function and use the stochastic optimization method proposed by [109] in this task. To train the entire framework containing joint representation learning and event detection, we have adopted the batch training approach with 64 sizes for 100 epochs. Similar to the

pre-training procedure, the initial learning rate is set to 0.1 and is declined at every 20 epoch step by multiplying 0.1. The learning rate decay is stopped when the learning rate reaches at 0.0001, and the rate is retained until the training finishes. In constructing the training dataset, to reduce an over-fitting of the network, one of the easiest and most common ways is the data augmentation which is to artificially enlarge the dataset using image transformation with label preserving [5]. We employ two simple approaches of the data augmentation, both of which produce artificial data from the original samples by applying simple image transformation.

The first strategy of data augmentation is generating the image translation with a probabilistic filter (e.g., Gaussian filter). We apply the Gaussian filter with multiple variances. The generated images are transformed images with different resolutions. This method increases the number of training samples by changing variance of the probabilistic filters. We extract 3D cubes from the raw input data and motion vector sequences, and smooth pixel values in 3D cubes using the filters.

The second strategy is applying horizontal reflections. By applying this strategy, we extract the fitted 3D cubes after finishing the first strategy. This method also increases the size of the training set. And, newly generated samples are highly independent on the original samples. Without these schemes, the proposed framework has difficulty handling the over-fitting problem, which would have forced us to designed much simpler network. Figure 3 shows the concept of the data augmentation used in this work.

3.4.4 Experiment

The proposed framework was evaluated on publicly accessible datasets which are UMN dataset [20], UCSD pedestrian dataset [62], and Subway dataset [18]. Three measurements are used to evaluate the performance of abnormal event detection: ROC, AUC, and EER. A high AUC value and low EER value indicates a better method to detect an abnormal event. The UMN and subway dataset are used to evaluate the performance of the proposed method for the frame-level abnormal event detection, and the UCSD dataset for both the LAEs and GAEs detections.Particularly, The results of the abnormal event detection are re-annotated to the event-level performance since the subway dataset only provides an event-level detection annotation. We have referred the evaluation scheme in Adam et al. [18].

For an efficient experiment, all frames are resized with 240×240 resolution. To construct the training dataset, we extract the motion and appearance 3D volumes various sizes and enlarge the dataset using aforementioned data augmentation approach. We extract 3 different-sized 3D volumes: 18×18, 20×20, and 25×25, and resize all them to 20×20. In evaluating phase, we extract the motion and appearance 3D volumes in uniform sizes with the overlapping rate 0.5. The 3D volumes of appearance and motion are $20 \times 20 \times 1 \times 5$ and $20 \times 20 \times 2 \times 5$. The 3D volume size is determined empirically. We referred to the various studies [20, 26, 104] for determining the volume size. The channels of each 3D volume are 1 and 2 since we used a grey scale image and dense optical flows, and we setted up 5 of temporal length empirically. We trained the proposed model using Adam stochastic optimization method [109] with the momentum of 0.9, and the weight decay of 0.0002. We set 0.5 probability for dropout for the three layers of the fully connected network. The proposed

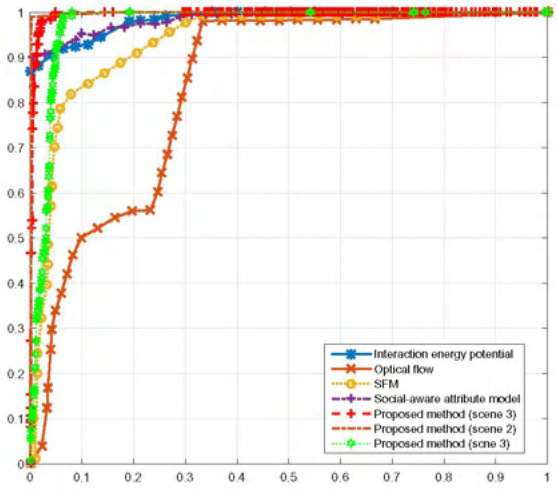


Figure 3.3: The ROCs for frame-level anomaly detection in the UMN dataset.

Methods	AUC
Optical flow	84%
Interaction energy potential [19]	98.5%
Social force model [20]	96%
Sparse reconstruction cost [26]	97.8%
Social-aware attribute [104]	98.6%
MIP-TS [28]	98.26%
Proposed method (Scene 1)	99.4%
Proposed method (Scene 2)	99.8%
Proposed method (Scene 3)	97.8%
Proposed method (Average)	**99%**

Table 3.2: The AUC values in the UMN dataset.

framework is implemented with the Tensorflow library of Google, and the experiments are carried out on a PC with a graphics card (NVIDIA GTX Titan X) and a multi-core 3.41 GHz GPU with 32 GB memory.

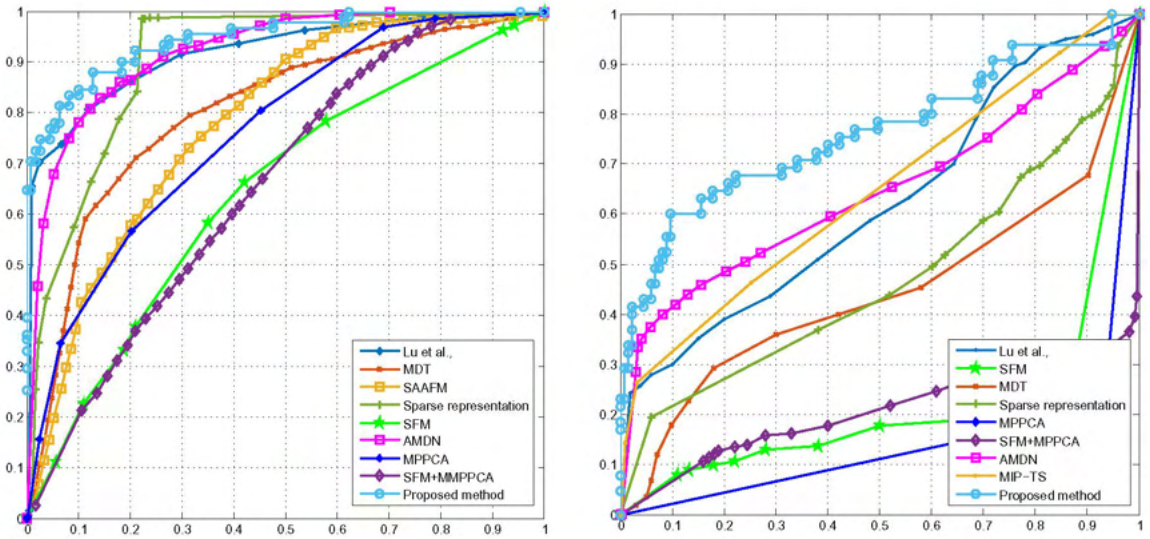
3.4.5 UMN dataset

We compare the proposed method to the method using general optical flow and the listed methods: Social Force Model (SFM) [20], Sparse reconstruction method [26], Social-aware attribute force model (SAAFM) [104], Interactive energy potential model [19], and the multi-scale motion interrelated pattern model [28]. We obtained an average AUC of 99% for UMN dataset. Table 3.2 shows the quantitative results of the proposed framework and other methods. ROC curves of the proposed framework and the other methods are illustrated in Figure 3.3. In the experiment using the UMN dataset, the proposed framework achieved detection results that are reasonably comparable with the other methods.

3.4.6 UCSD dataset

We test the performance of the proposed framework on the UCSD pedestrian dataset provided by [62].The ROC curves of the both the frame-level and pixel-level abnormal event detections for the Ped1 dataset are shown in Figure 3.4a and 3.4b. The ROC curves of the frame-level abnormal event detection for the Ped2 dataset is shown in Figure 3.5. We compare the event detection performance with that of several other methods: Social force model (SFM)[20], Mixture of Probabilistic Principal Component Analysis (MPPCA) [21] and its modified version with SFM, Mixture of Dynamic Texture (MDT) [62], Appearance and Motion DeepNet (AMDN) [35], social attribute-aware force model (SAAFM) [104], the method of Lu et al. [98], Motion Interpreted Patterns (MIP) [28], the method of Sabokrou et al. [110], the method proposed by Hasan et al. [37], Spaio-temporal auto-encoder (ST-Autoencoder) [36], and sparse representations [26]. Also, to demonstrate the structural efficiency of the proposed joint learning method, we compare the proposed method with the 3D convolutional neural network (Appearance 3D-ConvNet) based on the single input stream for appearance information inspired by Tran et al. [33] and Two stream 3D convolutional neural network (Two-stream 3D-ConvNet) [34]. In our experiment, Both networks are trained with the same parameter setting and training datasets to the proposed method.

Table 3.3 shows a quantitative comparison including EERs and AUCs for the proposed methods and the listed methods. Figure 3.6 shows the detection results of a local abnormal event using the proposed method in UCSD dataset. The experimental results show that the proposed framework can provide accurate and efficient abnormal event detection, and demonstrates that learning of joint spatio-temporal representation via the proposed framework



(a) (b)

Figure 3.4: The ROCs for frame-level abnormal event detection (a) and pixel-level abnormal event detection (b) in UCSD Ped1 dataset.

outperforms the listed state-of-the-art methods. We can observe that the AUC values of our framework on the ped1 and ped2 dataset are 94.4% and 94.8% respectively, which are higher than that of other comparison methods. Moreover, considering pixel-level evaluation for the anomaly localization, our method achieves the AUC value of 76.2%, which is significantly higher than the other methods. The gap between the proposed method and the best result among the list methods is 9%. The quantitative results in Table 3.3 demonstrate the advantages of the proposed joint learning method in the studies on the abnormal event detection. Specifically, the AUCs of the proposed method is larger than the appearance 3D-ConvNet [33] and the two-stream 3D-ConvNet [34]. This experimental results of the proposed method and two other methods could be interpreted as that even jointly learnt representation with simple hand-crafted feature can provide more discriminative power than the representation trained by the raw frames only.

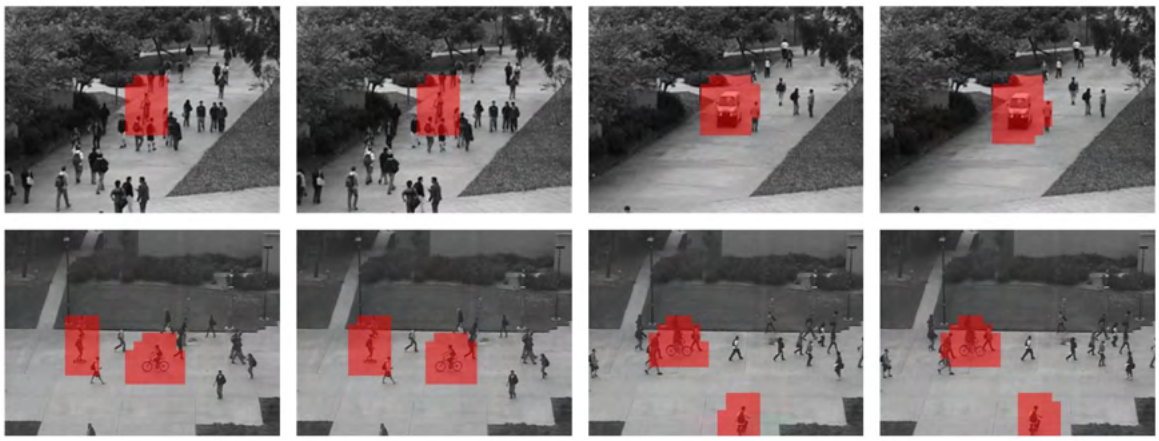


Figure 3.6: Detection results of a local abnormal event for UCSD datasets. The images in the first row are detection results in the Ped1 dataset, and the images in the second row show the detection results in the Ped2 dataset.

3.4.7 Subway dataset

In evaluation task using the subway dataset, we referred to the evaluation scheme in Lu et al. [98] for an efficient experiment. However, since abnormal event categories: '*No payment*' and '*Irregular interaction*', are not contained in '*Exit gate*' video. Therefore, we defined a new event category called '*Abnormal behaviours*' by grouping all events that labelled as '*Wrong direction*', '*No payment*', and '*Irregular interaction*'.

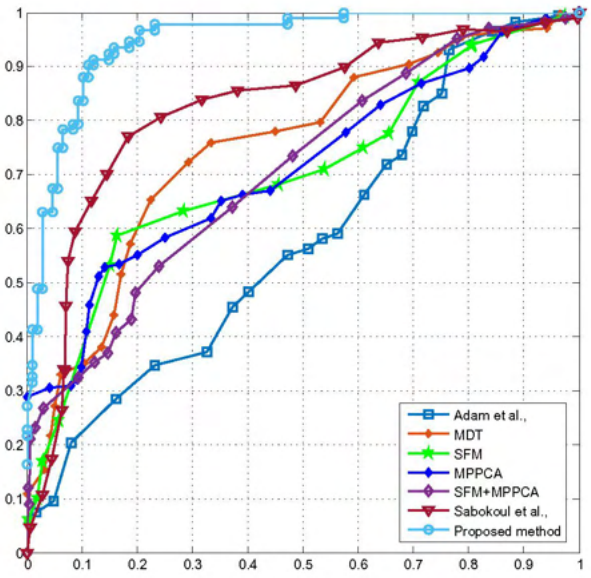


Figure 3.5: The ROCs for frame-level abnormal event detection in UCSD Ped2 dataset.

We constructed the training samples using the video sequences in the first 20 minutes and some part of video containing an abnormal event. Abnormal events in the training dataset are randomly picked from the given ground-truth, and we randomly selected 30% of abnormal

Methods	Ped1 (Frame)		Ped1 (Pixel)		Ped2 (Frame)	
	AUC	EER	AUC	EER	AUC	EER
MDT [62]	81.4%	25%	44.1%	58%	82.9%	25%
MIP-TS [28]	-	-	64.9%	41.3%	-	-
MPPCA [21]	59%	40%	20.5%	81%	69.3%	30%
MPPCA+SFM [21]	66.9%	32%	21.5%	72%	61.5%	35%
SFM [20]	67.5%	31%	19.27%	79%	55.6%	42%
SAAFM [104]	77.6%	29%	-	-	-	-
Sparse representation [26]	89.5%	19%	50.2%	53%	-	-
Lu et al., [98]	91.8%	15%	63.8%	59.1%	-	-
AMDN [35]	92.1%	16%	67.2%	40.1%	90.8%	17%
Sabokrou1 at al., [110]	-	-	-	-	82.4%	19%
Hasan et al., [37]	81.0%	27.9%	-	-	90.0%	21.7%
ST-Autencoder [36]	89.9%	12.5%	-	-	87.4%	12%
*Appearance 3D-ConvNet [33]	83.1%	25.34%	63.7%	55.24%	83.17%	26.4%
*Two-stream 3D-ConvNet [34]	86%	22%	64.4%	45%	83.2%	24%
Proposed method	**94.4%**	**12.8%**	**76.2%**	**31%**	**94.8%**	**11.1%**

Table 3.3: Quantitative performance comparision of different abnormal event detection methods using UCSD dataset. "-" means the results are not provided. "*" denotes the evaluation methods are implemented ourselves.

events contained in the ground-truth. The training dataset corresponding to each video is composed of 70000 normal 3D volumes and 4,0000 abnormal 3D volumes.

We compare the event detection performance of the proposed method with that of other methods: MPPCA [21], the method proposed by Lu et al. [98], sparse coding [23], sparse representation [26], and the method based on temporal regularization [37], and the spatio-temporal autoencoder [36]. Also, as the evaluation using UCSD dataset, we carry out additional experiments with general deep learning approaches [33, 34] for the action recognition and video analysis to demonstrate the methodological efficiency of the proposed method. Table 3.4 presents the comparison between the proposed methods and the other methods. Figure 3.7 shows the localization results of the detection results using the our method. The experimental

Figure 3.7: Detection results of abnormal events for the subway dataset. The three images on the left show the '*Loitering*' and the two images on the right presents '*Abnormal behaviour*'.

results for the subway dataset show that the number of events detected using the proposed method is larger than other methods. Also, the false alarm of the proposed method is a little bit lower than the listed methods. Interestingly, the experimental results using the proposed method, Appearance 3D-ConvNet, and Two-stream 3D-ConvNet, show that although the 3D-ConvNets can extract the features which can provide the spatial and temporal information simultaneously, the extracted features cannot provide sufficient discriminative guideline. Overall, experimental results for the subway dataset show that the proposed joint learning and detection methods allow us to discover a more discriminative feature than the existing methods.

3.4.8 Discussion

The experimental results show that the proposed method outperforms the state-of-the-art methods including the methods based on hand-crafted features and the deep learning based methods. Also, the experimental results of the proposed method and single and dual channel 3D-ConvNet methods show that although 3D-ConvNet can extract spatial and temporal representations simultaneously within the network naturally, the learnt features might enhance their discriminative powers which are obtained by the joint learning with simple hand-crafted

Methods	AB		LT		MISC		Total		FA	
	En	Ex	En	Ex	En	Ex	En	Ex	En	Ex
Ground-truth	**43**	**9**	**14**	**3**	**9**	**7**	**66**	**19**	**0**	**0**
Sparse coding [23]	38	9	14	3	8	7	60	19	5	2
MPCCA [21]	36	9	13	3	8	7	57	19	6	3
Sparse reconstruction [26]	27	9	-	-	-	-	-	-	4	2
Subspace [98]	30	6	9	3	7	5	46	14	7	4
Lu et al., [98]	36	9	13	3	8	7	57	19	4	2
Hasan et al., [37]	-	-	-	-	-	-	61	17	15	5
ST-Autoencoder [36]	-	-	-	-	-	-	61	18	9	10
*Appearance 3D-ConvNet [33]	31	9	14	3	8	6	53	18	11	4
*Two-stream 3D-ConvNet [34]	36	9	14	3	8	7	58	19	5	5
Proposed method	**40**	**9**	**14**	**3**	**7**	**7**	**61**	**19**	**4**	**2**

Table 3.4: Quantitative performance comparision of different abnormal event detection methods using the subway dataset. En and Ex denote '*Entrance video*' and '*Exit video*'. AB means abnormal behaviours; LT means loitering; MISC denotes misc; FA denotes false alarm. "-" means that the results are not provided. "*" denotes the evaluation methods are implemented ourselves.

features. Consequently, the key contribution of the proposed framework is that it can learn the joint spatio-temporal representation from 3D volumes of appearance and motion, without feature analysis, background subtraction, detection, or tracking methods. The proposed framework can detect abnormal events of diverse types that are defined by an appearance, a motion, or both within a single framework.

The main drawback is that the proposed method needs pre-defined parameters such as a threshold for detecting an abnormal event and a specific size of 3D volumes that are important things that related to the detection performance. In particular, determination of the size of 3D volumes is the critical problem in the studies for the volume-level abnormal event detection. An overly small volume cannot contain sufficient appearance and motion information to learn and analyze the normal or abnormal event patterns. On the other hand,

the too large volume might contain more information than you need so that it could converge to the poorly optimized solution during training a network. To solve this problem, we have constructed the training dataset by extracting the volume with various sizes. Additionally, the current framework can cause a computational inefficiency because of the separation of the event detection and localization tasks. However, these issues are general requirements for all volume-level anomaly detection methods. Furthermore, since the proposed framework is an off-line method, the proposed framework cannot ensure the reliable detection of abnormal events of a totally different types that are not included in training samples. In future work, we will devise a new model which can overcome the above mentioned drawbacks.

3.5 Adversarial Event Prediction (AEP)

The dominant approach to identifying abnormal events is deriving a model of normal events and then compute a likelihood or an error using the derived model. This methodology assumes that when the model takes an abnormal event sample as an input, the model produces either a lower likelihood or a larger error because the model is only trained by the normal event samples. This approach is represented as follows:

$$\text{AED}(x) = \begin{cases} \text{Abnormal}, & \text{if} \quad \mathcal{F}_{\boldsymbol{x}^n}(x) \leq \tau \\ \text{Normal}, & \text{Otherwise}, \end{cases}$$

where x is an input sample, $\mathcal{F}_{\boldsymbol{x}^n}$ is a model only trained with normal samples $\boldsymbol{x}^n \in \{x_1^n, x_2^n, x_3^n, ..., x_N^n\}$, and N is the number of normal event samples, and τ is a predetermined threshold.

In the above approach, the key components are the feature representation and the deriving a model for normal event samples. Numerous studies have been proposed to improve the robustness of these components [20, 104, 26, 13, 98]. Various hand-crafted features have proposed to improve the discriminative power in identifying an anomaly of events [20, 104, 26]. However, inherently, the performances of these approaches based on hand-crafted features are highly dependent on the parameter settings required for the features, and it is intractable to find an optimal parameter setting that can cover various environments accounted in the real world.

Being different from the above approaches based on hand-crafted features, several AED approaches based on deep learning methods are proposed [35, 111, 112, 113, 114, 115, 116, 117, 118]. These studies take advantage of the remarkable feature extraction capacity with the cascaded and weighted kernel structures of neural networks. Usually, these approaches are based on the reconstruction method inspired by autoencoder [119] and force it to minimize a reconstruction error in the training step [35, 110, 37, 112, 111, 120, 121]. These approaches assume that the error of abnormal events would be larger than the normal ones. These methods show remarkable achievements on AED than the conventional methods based on hand-crafted features.

Unfortunately, even though the deep learning-based approaches show remarkable performances, as uncertainty and complexity of feature distributions of various event patterns increases, it is inherently intractable to construct a model which can simultaneously take well-generalized for normal events and strictly-discriminative to abnormal events [112, 35]. Moreover, paradoxically, the superiority of deep learning, which is in learning of well-generalized

representation from a given dataset through minimizing of a reconstruction error, can be considered as a disadvantage to detect an anomaly of events, because the well-generalized mode can contain a potential risk which is producing smaller errors than the expectation for abnormal event samples [122, 123]. In other words, there is a possibility that a model trained with the reconstruction method cannot produce larger errors for abnormal events.

Several studies started to apply adversarial learning that does a model training using a classification in order to deal with this issue [124, 125, 122, 116, 126, 127, 120, 128]. Ravanbakhsh *et al.*,have presented an AED method with adversarial learning between the raw image and optical flow [124]. Liu *et al.*,[122] and Ionescu *et al.*,[116] address abnormal event detection using the classification. Liu have transferred AED as a future frame prediction problem and added extra cost functions for adversarial learning to train their model *et al.*,[125]. Liu *et al.*,have introduced the classification setting for AED [122]. Nawaratne *et al.*,have proposed Incremental Spatio-Temporal Learner (ISTL) in order to apply active learning strategy by utilizing an incremental learning for identifying abnormal events [127]. However, Ravanbakhsh *et al.*,[124] and Liu *et al.*,[125] need complementary information such as optical flow in order to improve the discriminative power of their model.

However, this can be problematic for hand-crafted features, which are intractable to find an optimal setting needed to be changed for various scene condition. Liu *et al.*,[122] and Ionescu *et al.*,[116] require samples of abnormal events to train their methods. Nawaratne *et al.*,[127] also need extra information such as optical flow, and particularly it requires iterative learning to apply active learning strategy. These methodological properties are probably an advantage in improving the discriminative power to exploit abnormal event samples randomly

selected from the testing dataset. However, it is perhaps impractical because of scarcity and unpredictability of abnormal events.

In this work, we propose Adversarial Event Prediction (AEP), a novel method to detect abnormal events based on event prediction which can improve AED performance without auxiliary information such as optical flow or prepared abnormal event samples. AEP initially derives the event prediction model for normal events based on the adversarial learning for predicting events' past and future. The proposed learning approach can provide an effective way to derive the model for normal events since it is based on the adversarial learning which can be thought as an effective strategy to avoid complex estimation for many intractable stochastic computations.

To detect abnormal events, AEP predicts future events and compare with given test samples. Since AEP is only trained with normal event samples, the prediction results would be inaccurate when AEP takes the samples containing abnormal events. Additionally, we employ a training in a random-matching manner in order to improve the robustness of AEP. We conducted experiments using UCSD-Ped dataset [62], CUHK Avenue dataset [98], Subway dataset [18], and UCF-Crime dataset [99] to demonstrate an efficiency of AEP for AED. The experimental results demonstrate that our AEP can outperform existing state-of-the-art methods.

The main contributions of our works are summarized as follows:

- A novel method for abnormal event detection (AED), called adversarial event prediction (AEP), which employs event prediction setting to identify an anomaly of events. AEP no requires extra features to model timescale and improve the discriminativeness of

learnt features.

- The adversarial learning for events' past and future to improve the robustness of the prediction model for abnormal event detection, which can provide discriminative representation learning without extra information such as optical flow.

- Extensive experimental results on AED, which contain both the performance analysis depending on the hyperparameter setting and the comprehensive comparison with the existing state-of-the-art methods including either GANs based methods or the methods using auxiliary features such as prepared anomaly samples optical flows in training their models.

3.5.1 Backgrounds on AEP

In this work, we formulate AED as an event prediction problem. Our hypothesis is that when the model takes an abnormal event sample as an input, the prediction results would be less accurate than the cases where the model takes normal event inputs if an event prediction model is trained only with normal event samples. This methodology for turning AED as the event prediction problem may look similar to the methods with the reconstruction [110, 112, 36, 35, 37]. The AED methods based on the reconstruction derive the mapping function $\mathcal{F}_{re} : \mathcal{X} \rightarrow \hat{\mathcal{X}}$, where $\mathcal{X}$ and $\mathcal{F}_{re}(\mathcal{X}) = \hat{\mathcal{X}}$ are the input samples and the reconstruction results, respectively. The mapping function is optimized by minimizing the reconstruction error $E(\mathcal{X}, \hat{\mathcal{X}})$, which is defined by Euclidean distances or stochastic difference measurements such as *Kullback-Leibler* divergence (KL-divergence). Similar to our hypothesis, these reconstruction-based methods assume that the methods produce larger

errors for abnormal event samples than those of the normal ones.

In these approaches based on the reconstruction, $\mathcal{F}_{re}$ is typically a bijective function and its co-domain $\hat{\boldsymbol{\mathcal{X}}} \in \{\mathcal{X}_i\}_{i=1:n}$, where n is the scale of the co-domain, would be equivalent to $\mathcal{X}$ when their methods are trained by minimizing $E(\mathcal{X}, \hat{\mathcal{X}})$. Intuitively, this functional property can be thought as a constraint in learning the mapping function by restricting a target of each input. It is essential to provide comprehensive dataset which can cover diverse visual patterns in order to improve the performance of the representation learning [5, 129]. Therefore, this property can negatively affect the capability of the representation learning of the mapping function.

Our insight is that the prediction can help to overcome this constraint since it can provide various pairs between inputs and the corresponding outputs by manipulating temporal intervals. Liu *et al.,* show that there is a possibility that the prediction setting can be used for improving the performance of AED [125]. The mapping function for event prediction setting is defined by, $\mathcal{F}_{ep} : \mathcal{X}^C \rightarrow \mathcal{X}^F$, where $\mathcal{X}^C$ denotes the current input and $\mathcal{X}^F$ indicate the corresponding future. The goal of $\mathcal{F}_{ep}$ is generating the accurate prediction results from given current samples $\mathcal{X}^c$. It may be similar to the learning objective of the methods based on the reconstruction, because it also can be interpreted as the error minimization between the given future samples and the prediction results. However, in contrast to the reconstruction setting which constrains inputs and outputs, the prediction setting is more flexible in learning the correlation between the input and output, because it is changeable to decide the output corresponded to each input by changing the time intervals between them.

Additionally to this, we exploit the adversarial learning to deal with AED transformed into

the prediction problem. As the AED problem setting is transformed to the prediction setting, the diversity of data, which can be used for training a model, would be greatly larger than the reconstruction setting, since the prediction setting can change the scale of the interval for assigning a corresponding future target for input data. As a result of increasing the diversity of the training data, it is essential to develop an approach for modelling normal events, which provide more strict discriminative abilities than reconstruction approaches. To address this issue, several studies [125, 130, 124, 131] have utilized adversarial learning methods as a complementary factor to improve the discriminative abilities of their normal event models based on a reconstruction setting.

3.5.2 Architectural details of Adversarial event prediction

Generally, generative adversarial learning can derive the complicated probabilistic distribution from a given dataset without complex stochastic approximation such as maximum likelihood estimation [12]. Therefore, several studies have exploited adversarial learning to improve the AED performance [124, 132, 125]. However, these approaches employ additional hand-crafted features such as optical flow to improve the discriminativeness of their AED models. This means that the approaches come with a drawback in finding an optimal hyperparameter for the hand-crafted features to achieve good AED performances for various scene conditions. Unlike these approaches, AEP improves the discriminative power to detect an anomaly of events by revising the cost function in a way that requires no extra features.

AEP is composed of three components: 1) Generator $\mathcal{G}$, 2) Latent feature discriminator $\mathcal{D}^L$, 3) Future discriminator $\mathcal{D}^F$, and 4) Past discriminator $\mathcal{D}^P$. Figure 3.8 illustrates the

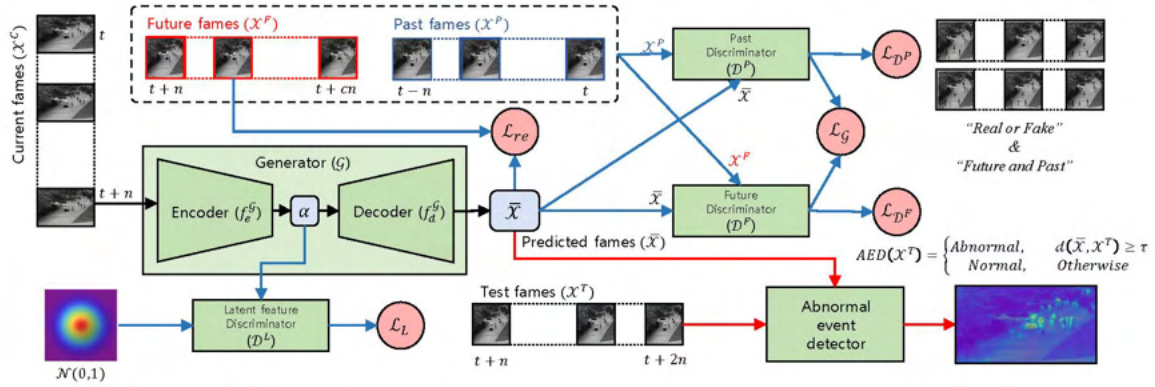


Figure 3.8: Structural details of the adversarial event prediction (AEP) based on adversarial learning for events' past $\mathcal{X}^P$ and future $\mathcal{X}^F$ for abnormal event detection (AED). The green objects denote the operational components e.g., the generator $\mathcal{G}$ and the discriminators $\mathcal{D}^F$, $\mathcal{D}^P$, and $\mathcal{D}^L$. The blue objects define the output of each component, and the red objects indicate the objective functions $\mathcal{L}_{\mathcal{D}^P}$, $\mathcal{L}_{\mathcal{D}^F}$ $\mathcal{L}_{\mathcal{D}^L}$, $\mathcal{L}_{\mathcal{G}}$ and $\mathcal{L}_{re}$. The solid black lines represent the workflow to generate the prediction results ($\bar{\mathcal{X}}$). The solid blue lines denote the workflow to compute the loss functions. The solid red lines show the process for AED using AEP.

structural details of AEP. The generator $\mathcal{G}$ is composed of the encoder $f_e^{\mathcal{G}}$ and the decoder $f_d^{\mathcal{G}}$, and it generates the prediction results $\bar{\mathcal{X}}$ using the current event sample $\mathcal{X}$. The generator $\mathcal{G}$ is composed of the encoder $f_e^{\mathcal{G}}$ and the decoder $f_d^{\mathcal{G}}$. The encoder $f_e^{\mathcal{G}}$ maps input samples into a latent plane: $f_e^{\mathcal{G}} : \mathcal{X} \rightarrow \alpha$. The decoder $f_d^{\mathcal{G}}$ generate the prediction results from the latent features : $f_d^{\mathcal{G}} : \alpha \rightarrow \bar{\mathcal{X}}$. The latent feature discriminator $\mathcal{D}^L$ derive the distribution of latent features α to the normal distribution. The future discriminator $\mathcal{D}^F$ and the past discriminator $\mathcal{D}^P$ distinguish that the generated samples are events' future or past.

The generator $\mathcal{G}$, the future discriminator $\mathcal{D}^F$, and the past discriminator $\mathcal{D}^P$ are built with 3D convolutional neural networks (3D-CNN) [102] and fully connected neural network (FC-NN). 3D-CNNs is employed to capture the spatial and temporal representation simultaneously from given event samples, and the FC-NN is used for abstracting the learnt representation extracted from 3D-CNNs. The network structure is inspired by various studies in video understanding [33, 133, 134, 31]. Figure 3.9 shows the details of the kernel dimensionalities

and connectivities for $f_e^{\mathcal{G}}$ and $f_d^{\mathcal{G}}$ in the generator $\mathcal{G}$ and the discriminators $\mathcal{D}^F$ and $\mathcal{D}^P$.

Generating prediction result using $\mathcal{G}$ is represented as follows:

$$\mathcal{G}(\mathcal{X}^C) = f_d^{\mathcal{G}} \cdot f_e^{\mathcal{G}}(\mathcal{X}^C) = \bar{\mathcal{X}}. \tag{3.13}$$

The outputs of the encoder $f_e^{\mathcal{G}}(\mathcal{X}^C) = \alpha$ are used as inputs of the latent feature discriminator $\mathcal{D}^L$, and the generated prediction results $\bar{\mathcal{X}}$ are directly applied to the two discriminators $\mathcal{D}^F$ and $\mathcal{D}^P$ to derive prediction model. $\mathcal{D}^F$ and $\mathcal{D}^P$ produce binary values representing a given input as events' certainties for future or past and fake or not, in order to compute a loss function for training AEP.

The future discriminator $\mathcal{D}^F$ is defined as follows,

$$\mathcal{D}^F(\mathcal{X}^{(*)_F}) = o^{\mathcal{D}^F}, o^{\mathcal{D}^F} \in R^1, \tag{3.14}$$

where $o^{\mathcal{D}^F}$ is the output of the future discriminator $\mathcal{D}^F$, and it is defined as a scalar value on Euclidean space. $\mathcal{X}^{(*)_F}$ denotes the the input of $\mathcal{D}^F$, and it can be regarded as $\mathcal{X}^F$ and $\bar{\mathcal{X}}$. $o^{\mathcal{D}^F}$ is used for distinguishing whether given samples are predicted results or the given ground-truth, by estimating the confidence value for distinguishing whether a given frame is the ground-truth.

$\mathcal{D}^P$ is structurally equal to $\mathcal{D}^F$, and it is defined by

$$\mathcal{D}^P(\mathcal{X}^{(*)_P}) = o^{\mathcal{D}^P}, o^{\mathcal{D}^P} \in R^1, \tag{3.15}$$

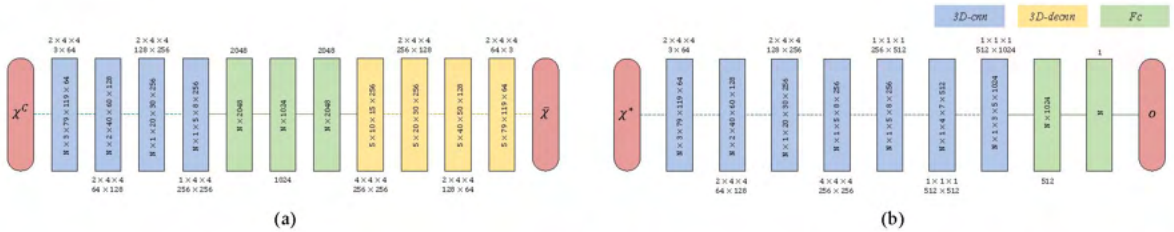


Figure 3.9: Structural details of the generator $\mathcal{G}$ and the two discriminators for events' past $\mathcal{D}^P$ and future $\mathcal{D}^F$. (a) denotes the structural details of the generator $\mathcal{G}$ and (b) indicates the structural details of the two discriminators $\mathcal{D}^F$ and $\mathcal{D}^P$. The red boxes indicate the inputs and the outputs of the generator and the discriminators. $\mathcal{X}^C$ and $\bar{\mathcal{X}}$ denote the current events and the prediction results respectively, and these are the input and the output of $\mathcal{G}$. $\mathcal{X}^*$ and o define the input and the output of $\mathcal{D}^F$ and $\mathcal{D}^P$. $\mathcal{X}^*$ could be defined by $\bar{\mathcal{X}}$, $\mathcal{X}^F$, and $\mathcal{X}^P$ depending on each discriminator. The blue boxes and the yellow boxes denote the 3D-convolutional neural network (*3D-cnn*) and the 3D-deconvolutional neural network (*3D-decnn*) respectively. The green boxes represent the fully-connected neural network (*Fc*). The figures presented in over and under the boxes show the dimensionalities of the network kernels on each layer. The figures inside of each box represent the dimensionalities of each layers' input or output.

where $o^{\mathcal{D}^P}$ is the output of the future discriminator $\mathcal{D}^P$. $\mathcal{X}^{(*)_P}$ indicates the input of $\mathcal{D}^P$ and $\bar{\mathcal{X}}$ and $\mathcal{X}^P$ are considered as the input of $\mathcal{D}^P$. These two discriminators are used for the adversarial learning for events' future and past to train the representation of AEP which can provide more discriminative power to predict events.

Additionally, the latent feature discriminator $\mathcal{D}^L$ is embedded to derive the distribution for the latent feature $p(\alpha)$. $\mathcal{D}^L$ is built with three FC-NNs with 2048, 2048, 1024 dimensionalities respectively. $\mathcal{D}^L$ is defined by

$$\mathcal{D}^L(f_e^{\mathcal{G}}(\mathcal{X}^C)) = \mathcal{D}^L(\alpha) = o^{\mathcal{D}^L}, o^{\mathcal{D}^L} \in R^1, \tag{3.16}$$

where $o^{\mathcal{D}^L}$ is the output of the discriminator $\mathcal{D}^L$, and it represents a confidence value for representing whether the input of $\mathcal{D}^L$ is actually generated from input samples $\mathcal{X}^C$ or randomly generated using the normal distribution $\mathcal{N}(0, 1)$. The adversarial learning applied to AEP is described in the next section.

3.5.3 Adversarial learning for past and future

The prerequisite for precise AED using AEP is deriving an optimal prediction model $p(\mathcal{X}^F|\mathcal{X}^C)$ with respect to the current event $\mathcal{X}^C$ and the corresponding future event $\mathcal{X}^F$. As the workflow of AEP mentioned in Section 3.5.2, the generator $\mathcal{G}$ initially maps the current frames $\mathcal{X}^C$ into the latent feature α through the encoder $f_e^{\mathcal{G}} : \mathcal{X} \rightarrow \alpha$, and generates the prediction results using the decoder $f_d^{\mathcal{G}} : \alpha \rightarrow \mathcal{X}^F$. The generator plays a role of the prediction model in the testing step. In this workflow, we assume that $p(\mathcal{X}^F|\alpha, \mathcal{X}^C) \approx p(\mathcal{X}^F|\alpha)$. Using above notations, $p(\mathcal{X}^F|\mathcal{X}^C)$ can be reformulated with $p(\alpha|\mathcal{X}^C)$ and $p(\mathcal{X}^F|\alpha, \mathcal{X}^C)$ as follows:

$$\begin{aligned} p(\mathcal{X}^F|\mathcal{X}^C) &= \int_\alpha \frac{p(\mathcal{X}^F, \mathcal{X}^C|\alpha)}{p(\mathcal{X}^C)} d\alpha \\ &= \int_\alpha p(\mathcal{X}^F|\mathcal{X}^C, \alpha) \frac{p(\alpha|\mathcal{X}^C)}{p(\alpha)} d\alpha \\ &\approx \int_\alpha p(\mathcal{X}^F|\alpha) \frac{p(\alpha|\mathcal{X}^C)}{p(\alpha)} d\alpha, \end{aligned} \tag{3.17}$$

where $p(\mathcal{X}^C)$ and $p(\alpha)$ denote the prior probabilities of $\mathcal{X}^C$ and α respectively. $p(\mathcal{X}^C)$ can be derived by given event samples defined as events' current. Consequently, to derive optimal $p(\alpha|\mathcal{X}^C)$, $p(\mathcal{X}^F|\alpha)$, and $p(\alpha)$ is necessary to attain the optimal $p(\mathcal{X}^F|\mathcal{X}^C)$, and we deal with this issue in term of an adversarial learning approach. The original intention of adversarial learning is to learn generative models while avoiding approximating many intractable probabilistic computations arising in other strategies *e.g.,* maximum likelihood estimation [12]. This intention is suitable to model $p(\mathcal{X}^F|\mathcal{X}^C)$ on AED to cover the various visual and kinetic patterns of normal events.

As we want to derive a robust mapping from current events to future events, the generated

Algorithm 1 Learning AEP by the proposed adversarial learning with events' past and future. η and λ_{re} denote the learning rates and the balancing weight of the reconstruction error $\mathcal{L}_{re}$, respectively. $\hat{\lambda}$ and $\ddot{\lambda}$ are predetermined balancing weight for the regularization term in the loss functions for the past and future discriminator.

Input: A current event sample $\mathcal{X}^C$ and corresponded samples for the event's past and future $\mathcal{X}^F$ and $\mathcal{X}^P$

Output: Updated parameters $\theta_{\mathcal{G}}, \theta_{\mathcal{D}^F}, \theta_{\mathcal{D}^P}$, and $\theta_{\mathcal{D}^L}$

for number of training iteration for each sample $\mathcal{X}_t^C$ **do**

• Produce the latent feature α and the prediction result $\bar{\mathcal{X}}$.

$\alpha = f_e^{\mathcal{G}}(\mathcal{X}^C)$

$\bar{\mathcal{X}} = \mathcal{G}(\mathcal{X}^C) = f^{\mathcal{G}_d} \cdot f^{\mathcal{G}_e}(\mathcal{X}^C)$

• Compute $\mathcal{L}_{\mathcal{D}^L}$ for the latent feature discriminator $\mathcal{D}^L$

$\mathcal{L}_{\mathcal{D}^L} = \mathbb{E}_{\mathcal{N}(0,1)}[\log(\mathcal{D}^L(\mathcal{N}(0,1)))] + \mathbb{E}_{\alpha \sim p_\alpha}[\log(1 - \mathcal{D}^L(\alpha))]$

• Compute $\mathcal{L}_{\mathcal{D}^F}$ for the future discriminator $\mathcal{D}^F$

$\mathcal{L}_{\mathcal{D}^F} = \mathbb{E}_{\mathcal{X}^C \sim p_{\mathcal{X}^C}}[\mathcal{D}^F(\mathcal{G}(\mathcal{X}^C))] - \mathbb{E}_{\mathcal{X}^F \sim p_{\mathcal{X}^F}}[\mathcal{D}^F(\mathcal{X}^F)] + \hat{\lambda}\mathbb{E}_{\hat{\mathcal{X}} \sim p_{\hat{\mathcal{X}}}}[(\|\nabla_{\hat{\mathcal{X}}}\mathcal{D}^F(\hat{\mathcal{X}})\|_2 - 1)^2]$

• Compute $\mathcal{L}_{\mathcal{D}^P}$ for the past discriminator $\mathcal{D}^P$

$\mathcal{L}_{\mathcal{D}^P} = \mathbb{E}_{\mathcal{X}^C \sim p_{\mathcal{X}^C}}[1 - \mathcal{D}^P(\mathcal{G}(\mathcal{X}^C))] = \mathbb{E}_{\mathcal{X}^P \sim p_{\mathcal{X}^P}}[1 - \mathcal{D}^P(\mathcal{X}^P)] + \ddot{\lambda}\mathbb{E}_{\ddot{\mathcal{X}} \sim p_{\ddot{\mathcal{X}}}}[(\|\nabla_{\ddot{\mathcal{X}}}\mathcal{D}^P(\ddot{\mathcal{X}})\|_2^2)]$

• Compute $\mathcal{L}_{re}$ and $\mathcal{L}_{\mathcal{G}^*}$ for the generator $\mathcal{G}$

$\mathcal{L}_{\mathcal{G}} = \mathbb{E}_{\mathcal{X}^C \sim p_{\mathcal{X}^C}}[\mathcal{D}^F(\mathcal{G}(\mathcal{X}^C))] + \mathbb{E}_{\mathcal{X}^C \sim p_{\mathcal{X}^C}}[1 - \mathcal{D}^P(\mathcal{G}(\mathcal{X}^C))] + \mathbb{E}_{\alpha \sim p_\alpha}[\log(1 - \mathcal{D}^L(\alpha))]$

$\mathcal{L}_{re} = \mathbb{E}_{\mathcal{X}^C, \mathcal{X}^F}\|\mathcal{X}^F - \mathcal{G}(\mathcal{X}^C)\|_2^2$

$\mathcal{L}_{\mathcal{G}^*} = \mathcal{L}_{\mathcal{G}} + \lambda_{re}\mathcal{L}_{re}$

• Update the parameters $\theta_{\mathcal{G}}, \theta_{\mathcal{D}^F}, \theta_{\mathcal{D}^P}$, and $\theta_{\mathcal{D}^L}$

$\theta_{\mathcal{D}^F} \leftarrow \theta_{\mathcal{D}^F} + \eta \frac{d\mathcal{L}_{\mathcal{D}^F}}{d\theta_{\mathcal{D}^F}}$

$\theta_{\mathcal{D}^P} \leftarrow \theta_{\mathcal{D}^P} + \eta \frac{d\mathcal{L}_{\mathcal{D}^P}}{d\theta_{\mathcal{D}^P}}$

$\theta_{\mathcal{D}^L} \leftarrow \theta_{\mathcal{D}^L} + \eta \frac{d\mathcal{L}_{\mathcal{D}^L}}{d\theta_{\mathcal{D}^L}}$

$\theta_{\mathcal{G}} \leftarrow \theta_{\mathcal{G}} + \eta \frac{d\mathcal{L}_{\mathcal{G}^*}}{d\theta_{\mathcal{G}}}$

end for

return Updated parameters $\theta_{\mathcal{G}}, \theta_{\mathcal{D}^F}, \theta_{\mathcal{D}^P}$, and $\theta_{\mathcal{D}^L}$

events should be consistent with the frames assigned as the prediction target. The general adversarial loss in Eq. 2.5 is unsuitable for applying the time-series data, since it has a problem in optimizing the model by presenting inconsistent training process even when a model is trained with still image. AEP employs the WGAN-GP [12], which improves the stability in learning generative adversarial network (GAN).

The proposed adversarial learning for past events and future events is inspired by metric learning, *e.g.* triplet loss [135, 136, 137] and quadruple loss [138, 139]. In metric leaning, it

is a commonly used approach to use both positive and negative samples in learning a model, in order to improve the discrminative aspect of learnt features. The future event samples $\mathcal{X}^F$ play as a role for the positive samples, and the role of the past event samples $\mathcal{X}^P$ is a negative sample in the metric learning. By using the proposed adversarial learning for events' future and past, AEP can provide more discriminative representation for predicting the event's future by constraining the representation learning for past events.

The loss function for the future discriminator on the proposed adversarial learning are defined by,

$$\begin{aligned}\mathcal{L}_{\mathcal{D}^F} = & \mathbb{E}_{\mathcal{X}^C \sim p_{\mathcal{X}^C}}[\mathcal{D}^F(\mathcal{G}(\mathcal{X}^C))] \\ & - \mathbb{E}_{\mathcal{X}^F \sim p_{\mathcal{X}^F}}[\mathcal{D}^F(\mathcal{X}^F)] \\ & + \hat{\lambda}\mathbb{E}_{\hat{\mathcal{X}} \sim p_{\hat{\mathcal{X}}}}[(\|\nabla_{\hat{\mathcal{X}}}\mathcal{D}^F(\hat{\mathcal{X}})\|_2 - 1)^2],\end{aligned} \tag{3.18}$$

where $\mathcal{X}^C$ is the current events which we want to predict the future, $\mathcal{X}^F$ denotes the future events assigned as a prediction target corresponding to the current frames. The third term in Eq. 3.18 plays as a role of a regularizer on computing the gradient of the loss function with the balancing weight $\hat{\lambda}$, and it is computed with $\mathcal{G}(\mathcal{X}^C)$, $\mathcal{X}^F$, and $t \in [0, 1]$ as follows,

$$\hat{\mathcal{X}} = t\mathcal{G}(\mathcal{X}^C) + (1 - t)\mathcal{X}^F. \tag{3.19}$$

The loss of the past discriminator with the past events $\mathcal{X}^P$ is defined as follows,

$$\begin{aligned}\mathcal{L}_{\mathcal{D}^P} &= \mathbb{E}_{\mathcal{X}^C \sim p_{\mathcal{X}^C}}[1 - \mathcal{D}^P(\mathcal{G}(\mathcal{X}^C))] \\ &= \mathbb{E}_{\mathcal{X}^P \sim p_{\mathcal{X}^P}}[1 - \mathcal{D}^P(\mathcal{X}^P)] \\ &+ \ddot{\lambda}\mathbb{E}_{\ddot{\mathcal{X}} \sim p_{\ddot{\mathcal{X}}}}[(\|\nabla_{\ddot{\mathcal{X}}}\mathcal{D}^P(\ddot{\mathcal{X}})\|_2^2)].\end{aligned} \tag{3.20}$$

The second term is the regularization term with $\mathcal{G}(\mathcal{X})$, $\mathcal{X}^P$, and balancing weight $\ddot{\lambda}$, computed by Eq. 3.19, and it is defined as follows,

$$\ddot{\mathcal{X}} = t\mathcal{G}(\mathcal{X}^C) + (1-t)\mathcal{X}^P. \tag{3.21}$$

In optimizing AEP with these two loss functions $\mathcal{L}_{\mathcal{D}^F}$ and $\mathcal{L}_{\mathcal{D}^P}$ for the future discriminator $\mathcal{D}^F$ and the past discriminator $\mathcal{D}^P$, The generator $\mathcal{G}$, generates the prediction results $\bar{\mathcal{X}}$, while the other, the discriminators, evaluate them for authenticity. For instance, the future discriminator decides whether each sample that it reviews belongs to the actual future event samples or the predicted results, and the past discriminator determines whether each sample that it review belongs to the actual past event samples or the prediction results.

Intuitively, the difference between $\mathcal{L}_{\mathcal{D}^F}$ and $\mathcal{L}_{\mathcal{D}^P}$ is an objective of each loss. The objective of $\mathcal{L}_{\mathcal{D}^F}$ is maximizing $\mathcal{D}(\mathcal{X}^F)$. On the other hands, the objective of $\mathcal{L}_{\mathcal{D}^P}$ is an opposite to $\mathcal{L}_{\mathcal{D}^F}$, and it leads to $\mathcal{D}(\mathcal{X}^P)$ to zero. This approach is similar to Ying *et al.*,[140] applying an additional term for minimizing the likelihood of noisy information in order to achieve better performance in training GAN. Consequently, using these two discriminator losses can improve the robustness of $p(\mathcal{X}^F|\mathcal{X}^C)$ by maximizing a likelihood for correct prediction and

minimizing a likelihood for the wrong answer.

Additionally, we add additional loss for the latent feature discriminator $\mathcal{D}^L$ for deriving precise $p(\alpha)$ as follows,

$$\begin{aligned}\mathcal{L}_{\mathcal{D}^L} &= \mathbb{E}_{z\sim\mathcal{N}(0,1)}[\log(\mathcal{D}^L(z))] \\ &+ \mathbb{E}_{\alpha\sim p_\alpha}[\log(1-\mathcal{D}^L(\alpha))],\end{aligned} \tag{3.22}$$

where α denotes the output of encoder $f^{\mathcal{G}_e}$ on the generator $\mathcal{G}$. $\mathcal{D}^L$ aims to distinguish between the encoding produced by $f^{\mathcal{G}_e}$ and the prior normal distribution. In learning AEP, $\mathcal{L}_{\mathcal{D}^L}$ tries to encode $\mathcal{X}^c$ to α with distribution close to $\mathcal{N}(0,1)$.

The loss function for the generator $\mathcal{G}$ is defined by

$$\begin{aligned}\mathcal{L}_{\mathcal{G}} &= \mathbb{E}_{\mathcal{X}^C\sim p_{\mathcal{X}^C}}[\mathcal{D}^F(\mathcal{G}(\mathcal{X}^C))] \\ &+ \mathbb{E}_{\mathcal{X}^C\sim p_{\mathcal{X}^C}}[1-\mathcal{D}^P(\mathcal{G}(\mathcal{X}^C))] \\ &+ \mathbb{E}_{\alpha\sim p_\alpha}[\log(1-\mathcal{D}^L(\alpha))].\end{aligned} \tag{3.23}$$

In addition to the adversarial learning for events' past and future, similar to the studies [140, 141, 125, 112, 124, 132, 142, 120], we adopt the reconstruction error to optimize the generator $\mathcal{G}$. It is a common way to enforce the output of the generator to be close to the target through the minimization of the reconstruction error based on the pixel-wise mean square error (MSE). It is calculated in the form

$$\mathcal{L}_{re} = \mathbb{E}_{\mathcal{X}^C,\mathcal{X}^F}\|\mathcal{X}^F - \mathcal{G}(\mathcal{X}^C)\|_2^2. \tag{3.24}$$

Consequently, the loss function for optimizing the generator $\mathcal{G}$ is defined by,

$$\mathcal{L}_{\mathcal{G}^*} = \mathcal{L}_{\mathcal{G}} + \lambda_{re}\mathcal{L}_{re}, \tag{3.25}$$

where λ_{re} indicates the hyperparameter to take the weight for the reconstruction loss.

Given the definition of above loss functions, each discriminator and the generator are trained by maximizing or minimizing the corresponding loss function, and these are represented as follows,

$$\theta_{\mathcal{D}^F} = \theta_{\mathcal{D}^F} + \eta\frac{d\mathcal{L}_{\mathcal{D}^F}}{d\theta_{\mathcal{D}^F}}, \quad \theta_{\mathcal{D}^P} = \theta_{\mathcal{D}^P} + \eta\frac{d\mathcal{L}_{\mathcal{D}^P}}{d\theta_{\mathcal{D}^P}},$$
$$\theta_{\mathcal{D}^L} = \theta_{\mathcal{D}^L} + \eta\frac{d\mathcal{L}_{\mathcal{D}^L}}{d\theta_{\mathcal{D}^L}}, \quad \theta_{\mathcal{G}} = \theta_{\mathcal{G}} + \eta\frac{d\mathcal{L}_{\mathcal{G}}}{d\theta_{\mathcal{G}}},$$

where θ^* denotes the parameters corresponded to the generator $\mathcal{G}$ and the discriminators $\mathcal{D}^L, \mathcal{D}^F$ and $\mathcal{D}^P$ on AEP. Algorithm 1 describes how to learn AEP by the proposed adversarial learning method. In our experiment, we demonstrate that the proposed adversarial losses play a vital role in deriving more optimal $P(\mathcal{X}^F|\mathcal{X}^C)$.

3.5.4 Multi-target random-matching

Additionally, we apply the random-matching manner to train AEP. A constant-matching method, which fixes a time-interval between an input data and future target data, have been being utilized in spatio-temporal feature modeling for various visual analysis such as abnormal event detection [35, 125, 20, 37], action recognition [38, 102], and video generation [143, 144]. However, we employ a random-matching manner to improve the robustness of the

model. The random-matching selects future data with the randomly assigned time-interval. Figure 3.10 illustrates the conceptual comparison between the constant-matching and the random-matching on a time-series data. In constant matching (Figure 3.10(a)), each $\mathcal{X}^C$ has a corresponding $\mathcal{X}^F$, and the intervals ϵ_* between $\mathcal{X}^C$ and $\mathcal{X}^F$ are all equivalent as follows $\epsilon_t \equiv \epsilon_{t+1}$. In contrast to the constant-matching, the intervals of random-matching (Figure 3.10(b)) is changeable. Therefore it is possible to assign diverse targets to the input data in training models. Intuitively, improving the diversity of the training data can affect the representation learning performance of a given model.

Moreover, we revise the objective function utilizing the random-matching with respect to the multiple prediction targets. As shown in Figure 3.10, the random-matching has a potential that a model can accomodate various temporal intervals in one training step. Eq.3.18 is reformulated to process the multiple targets $\boldsymbol{\mathcal{X}}_t^{F_{1:n}} = \{\mathcal{X}_t^{F_i}\}_{i=1:n}$, where n is the number of randomly picked targets, for the input $\mathcal{X}_t^C$, as follows:

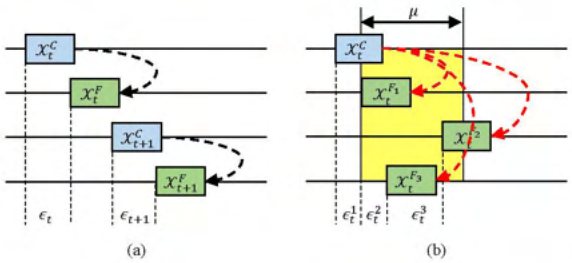


Figure 3.10: Comparison between the concept of (a) constant matching and (b) random-matching in learning AEP. The blue and green boxes denote the input samples $\mathcal{X}^C$ and the corresponding prediction target $\mathcal{X}^F$ respectively. The black and red dotted lines indicate the connections between $\mathcal{X}^C$ and $\mathcal{X}^F$. The yellow range defined by μ represents the interval for generating the prediction target randomly. ϵ_t indicates the t^{th} interval between $\mathcal{X}_t^C$ and $\mathcal{X}_t^F$. In random-matching, various prediction targets $\boldsymbol{\mathcal{X}}_t^{F_{1:N}} = \{\mathcal{X}^{F_t^i}\}_{i=1:N}$, where N is the number of randomly picked targets, can be selected.

$$
\begin{aligned}
\mathcal{L}_{\mathcal{D}^F}^{\text{MTRM}} &= \mathbb{E}_{\mathcal{X}_t^C \sim p_{\mathcal{X}^C}}[\mathcal{D}^F(\mathcal{G}(\mathcal{X}_t^C))] \\
&- \frac{1}{n}\sum_{i=1}^{n}[\mathbb{E}_{\mathcal{X}^{F_i} \sim p_{\mathcal{X}^{F_i}}}[\mathcal{D}^F(\mathcal{X}^{F_i})]] \\
&+ \frac{\hat{\lambda}}{n}\sum_{i=1}^{n}[\mathbb{E}_{\hat{\mathcal{X}}^i \sim p_{\hat{\mathcal{X}}^i}}[(\|\nabla_{\hat{\mathcal{X}}^i}\mathcal{D}^F(\hat{\mathcal{X}}^i)\|_2 - 1)^2]].
\end{aligned}
\tag{3.26}
$$

This matching manner can be easily extended to the loss function for the past discriminator (Eq. 3.20) and the reconstruction loss (Eq. 3.24) as well. Given randomly picked past event samples $\boldsymbol{\mathcal{X}}_t^{P_{1:n^P}} = \{\mathcal{X}_t^{P_i}\}_{i=1:n^P}$, where n^P is the number of randomly picked targets for events' past, Eq. 3.20 applying the random-matching manner is defined by

$$\begin{aligned}\mathcal{L}_{\mathcal{D}^P}^{\text{MTRM}} = \frac{1}{n}\sum_{i=1}^{n^P}\mathbb{E}_{\mathcal{X}^{P_i}\sim p_{\mathcal{X}^{P_i}}}[1-\mathcal{D}^P(\mathcal{X}^{P_i})] \\ + \frac{\ddot{\lambda}}{n}\sum_{i=1}^{n^P}\mathbb{E}_{\ddot{\mathcal{X}}^i\sim p_{\ddot{\mathcal{X}}^i}}[(\|\nabla_{\ddot{\mathcal{X}}^i}\mathcal{D}^P(\ddot{\mathcal{X}}^i)\|_2^2)].\end{aligned} \tag{3.27}$$

The reconstruction loss processing randomly picked future event samples $\boldsymbol{\mathcal{X}}_t^{F_{1:n}}$ are defined by

$$\mathcal{L}_{re}^{\text{MTRM}} = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{\mathcal{X}^C,\mathcal{X}^{F_i}}\|\mathcal{X}^{F_i}-\mathcal{G}(\mathcal{X}^C)\|_2^2. \tag{3.28}$$

To demonstrate an efficiency of the random-matching in processing, we train AEP with different objective functions based on the four matching ways: 1) single-target constant-matching (STCM), 2) the single-target random-matching (STRM), 3) the multi-target constant-matching (MTCM) and 4) the multi-target random-matching (MTRM). The experimental results shown in Section 3.5.2 include the AED performance comparison depending on the matching manners.

3.5.5 Abnormal event detection and localization

AED based on AEP is straightforward. As shown in Figure 3.8, after AEP training has been completed, the two discriminators $\mathcal{D}^F$ and $\mathcal{D}^P$ are not utilized for the further step for

detecting and localizing abnormal events. The generator $\mathcal{G}$ predicts the future frames $\bar{\mathcal{X}}$ using a given frames $\mathcal{X}^C$. AEP employs constant matching in the test step. Future frame prediction using AEP is inherently a stochastic problem, since AEP is only optimized to normal event frames when AEP take abnormal frame as an input, the prediction results probably would be poor than normal input. AEP can detect abnormal events by comparing the predicted results $\bar{\mathcal{X}}$ the corresponding frames $\mathcal{X}^T$ in the test step. However, precisely localizing abnormal events can not be achieved by just computing the likelihood for the prediction results. To deal with this issue, we employ a sliding window technique for the localization of abnormal events. AEP can localize abnormal events by comparing the predicted results $\bar{\mathcal{X}}$ the given test frames $\mathcal{X}^T$ based on the sliding window approach.

To compare the two frames $\bar{\mathcal{X}}$ and $\mathcal{X}^T$, we formulate a distance metric based on Jeffrey divergence, which is a modified KL-divergence to take symmetric property. According to Rubner *et al.,*[145], Euclidean distances such as $l1$-norm and $l2$-normal are not suitable as a similarity metric for images since neighboring values are not considered. Jeffrey divergence is numerically stable, symmetric, and invariant to noise and input scale [146]. The distance metric based on Jeffrey divergence is defined as follows.

$$
\begin{aligned}
d(\bar{\mathcal{X}}, \mathcal{X}^T) &= \sum_{i,j} (\bar{x}_{i,j} \log \frac{\bar{x}_{i,j}}{m_{i,j}} - x^T_{i,j} \log \frac{x^T_{i,j}}{m_{i,j}}), \\
m_{i,j} &= \frac{\bar{x}_{i,j} + x^T_{i,j}}{2},
\end{aligned}
\tag{3.29}
$$

where $\bar{x}_{i,j}$ and $x^T_{i,j}$ denote the the pixel value of the coordinate i, j on the predicted frames $\bar{\mathcal{X}}$ and the frames as the comparison target $\mathcal{X}^T$ respectively.

As we employ the sliding-window approach, computing abnormality is conducted with the small-size 3D window. Therefore, the localization of abnormal events is also simple. As a result of computing the distance between $\bar{\mathcal{X}}$ and $\mathcal{X}^T$ are carried out using a small template, we can distinguish that whether each template contains an abnormal event or not.

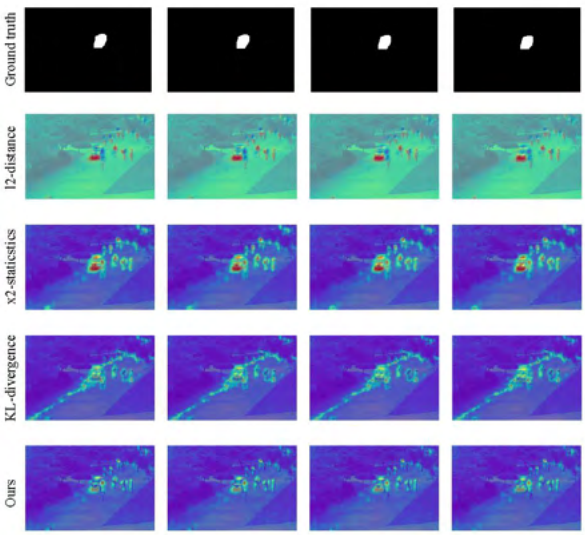


Figure 3.11: The visualization results of event abnormalities using UCSDPed1 dataset, depending on the abnormality metrics: $l2$-distance, $x2$-statistics, *Kullback-Leibler* divergence, and the our metric (Eq. 3.29). The graph of each plot shows the trend of abnormality with respect to the time-sequence and the images shows the visualization results of event abnormalities.

The simple experiments to compare the discriminative powers of abnormality metrics are conducted. Based on the prediction results $\bar{\mathcal{X}}$ and the test frames $\mathcal{X}^T$, we compute abnormality using our metric and three additional distance metrics: 1) $l2$-distance: $\sqrt{\sum_{ij}(\bar{x}_{ij} - x^T_{ij})}$, 2) $x2$-statistics: $\sum_{ij} \frac{(\bar{x}_{ij} - m_{ij})^2}{m_{ij}}$, where $m_{ij} = \frac{\bar{x}_{ij} + x^T_{ij}}{2}$, and 3) KL-divergence: $\sum_{ij} \bar{x}_{ij} \log \frac{\bar{x}_{ij}}{x^T_{ij}}$. Figure 3.11 represents the visualization of events' anomaly by projecting the abnormality measurement results on corresponding frames. As shown in Figure 3.11, the proposed metric spots the areas of abnormal events more discriminatively than the other metrics. The results using $l2$-distance and $x2$-statistics show a similar trend on the localization of AED results to the results using our metric. The results using KL-divergence produce many incorrect detection outcomes.

The visualization results can be interpreted as follows. According to Tomasi *et al.*,[145], the distance metrics using $l2$-distance and $x2$-statistics cannot consider the neighboring

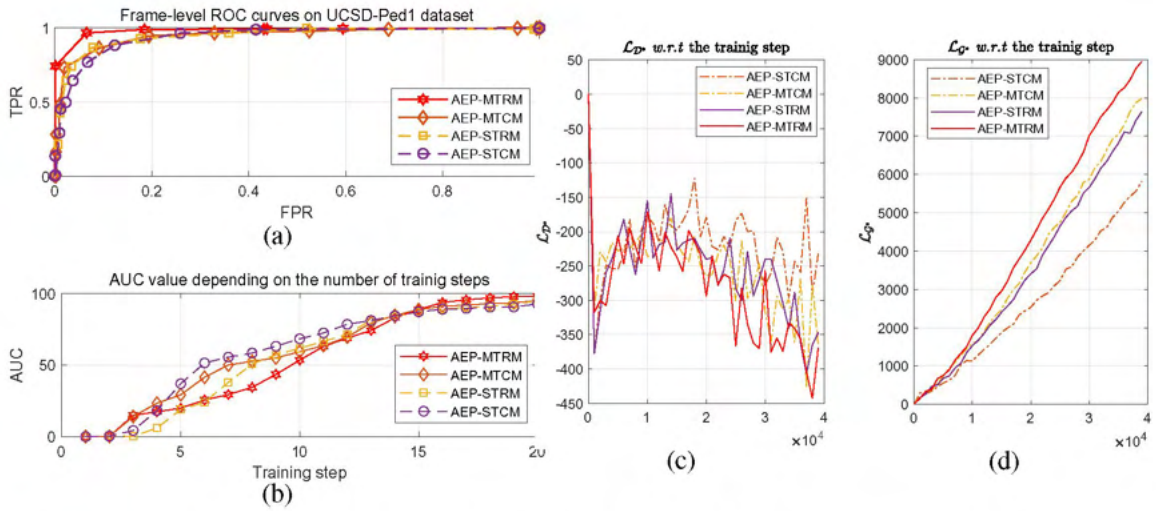


Figure 3.12: Graphical results for performance analysis of AEPs according to the four matching manners. (a) contains the ROC curves of AEPs trained by *'single target constant matching'* (STCM), *'multi-target constant matching'* (MTCM), *'single target random matching'* (STRM), and *'multi-target random matching'* (MTRM). (b) shows the trend of AEPs' AUCs, according to the matching manners, with respect to the number of training step. (c) and (d) represents the trends of $\mathcal{L}_{\mathcal{D}^*}$ and $\mathcal{L}_{\mathcal{G}^*}$ according to the number of training step, respectively. The AUC values on (b) are recorded by every 2K training steps, and $\mathcal{L}_{\mathcal{D}^*}$ of (c) and $\mathcal{L}_{\mathcal{G}^*}$ of (d) are recorded by every 10 training steps. These results are produced based on UCSD-Ped1 dataset, and the ROC curves and the AUC values are produced based on the frame-level evaluation.

properties, and it is a disadvantage in comparing two high-dimensional objects. Particularly, anomaly of events can appear with a diverse variation of motion and appearance. Therefore, simply comparing the vectorial distance is probably unsuitable for comparing the difference between two objects for AED. Consequently, based on the distance computation results for each template, we can localize areas of abnormal events.

3.5.6 Experiments

The frame-level and pixel-level measurements are exploited to evaluate AED performance. Additionally, we compute the area under curve (AUC) and the equal error rate (EER) to provide a quantitative comparison.

We carry out the experiment for comparison of AED performance depending on the matching manners and whether the past discriminator $\mathcal{D}^P$ is applied. UCSD-Ped dataset is used for this experiment. The hyperparameter settings for training the models are all the same, and only matching manners are different. AUC and EER are computed for quantitative comparison. Not only quantitative results are shown but also we analyze the trends of the discriminator losses $\mathcal{L}_{\mathcal{D}^{\mathcal{F}}}$, $\mathcal{L}_{\mathcal{D}^{\mathcal{P}}}$, and $\mathcal{L}_{\mathcal{D}^{\mathcal{L}}}$, and the generator loss $\mathcal{L}_{\mathcal{G}^*}$ during training the AEPs. For efficiency of experiments, we define the total loss which is a summation of the losses of the three discriminators as follows,

$$\mathcal{L}_{\mathcal{D}^*} = \mathcal{L}_{\mathcal{D}^{\mathcal{F}}} + \mathcal{L}_{\mathcal{D}^{\mathcal{P}}} + \mathcal{L}_{\mathcal{D}^{\mathcal{L}}}.$$

The AUCs and EERs depending on the matching manners are contained in Table 3.5. The experimental results demonstrate that applying the past discriminator $\mathcal{D}^P$ can improve AED performance. The AED performances applying $\mathcal{D}^p$ generally achieves better performances than the others. Additionally, among the results achieved by the AEPs applying $\mathcal{D}^P$, the AEP trained by MTRM achieve the best performances on UCSD-Ped dataset, compared to the AEPs, which are trained with STCM, produces the lowest performance for all experiments. AEP_{MTRM} produces AUC of 97.92 and EER of 6.07 for the frame-level evaluation, and achieves AUC of 74.83 and EER of 31.06 for the pixel-level evaluation on UCSD-Ped1 dataset. It achieves AUC of 97.31 and EER of 7.52 for the frame-level evaluation on UCSD-Ped2 dataset. Each AUC result shows at least 2% of improvement than the second-ranked results. The poorest results are achieved by AEP_{STCM}. The evaluation results of AEP_{STRM} on

Methods	Ped1 (Frame)		Ped1 (Pixel)		Ped2 (Frame)	
	AUC	EER	AUC	EER	AUC	EER
Training without the discriminator for events' past $\mathcal{D}^P$						
AEP_{STCM}	87.97	16.51	64.31	46.51	75.08	16.52
AEP_{MTCM}	94.14	11.09	69.08	39.36	91.12	10.72
AEP_{STRM}	94.52	8.61	72.99	34.01	92.43	10.09
AEP_{MTRM}	96.51	9.26	71.75	38.16	95.19	9.15
Training with the discriminator for events' past $\mathcal{D}^P$						
AEP_{STCM}	92.61	13.72	69.83	41.72	80.54	16.78
AEP_{MTCM}	95.09	7.94	73.95	35.12	94.92	10.71
AEP_{STRM}	95.12	7.31	72.51	34.61	95.02	9.1
AEP_{MTRM}	**97.92**	**6.07**	**74.83**	**31.06**	**97.85**	**7.52**

Table 3.5: Quantitative performance comparison of the AED performance on AEPs using UCSD-Ped dataset depending on applying the past discriminator $\mathcal{D}^P$ and the matching manner. The bolded figures indicate the best performances for each evaluation. 'STCM', 'MTCM', 'STRM', and 'MTRM' denotes each model is trained with *'single-target constant matching'*, *'multi-target constant matching'*, *'single target random-matching'*, and *'multi-target random-matching'*.

UCSD-Ped1 dataset, produce AUC of 92.61 and EER of 13.72 to the frame-level evaluation, and achieves AUC of 69.83 and EER of 41.72 to the pixel-level evaluation. Also, it achieves the AUC of 97.85 and the EER of 7.52 to the frame-level evaluation on UCSD-Ped2 dataset.

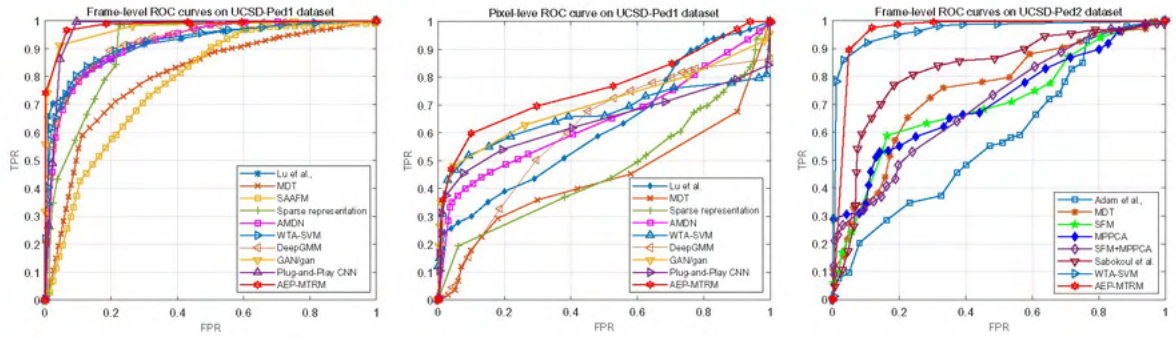
In addition to the quantitative results, the ROC curves and the graph of the trend for AUC with respect to the training step also show that the random-matching manner can improve the AED performance of AEP. Figure 3.12(a) and Figure 3.12(b) show the ROC curves for the frame-level evaluation, and the trend of AUC value with respect to the number of training steps, on UCSD-Ped1 dataset, respectively. As similar as the quantitative results using AUC and EER, graph analysis using ROC curves also presents a similar result which is that AEP trained with random-matching manners achieve better performance than the others. As shown in Figure 3.12(a), the ROC curve of AEP_{MTRM} has the steepest gradient. The ROC curves of AEP_{STRM} and AEP_{MTCM} have a similar trend. The worst ROC curve is produced by AEP_{STCM}.

However, there is a trade-off between the AED performance and the convergence speed of training AEPs employing random-matching manners. As shown in Figure 3.12(b), AEP_{STCM} shows the fastest speed in increasing of AUC in the early step for the training, and this trend is preserved until the number of the training step is over than 30K. This circumstance is also observed in the trends of the discriminator loss and generator loss. Figure 3.12(c) and Figure 3.12(d) contains the trend of the loss $\mathcal{L}_{\mathcal{D}^*}$ for the discriminators $\mathcal{D}^F$ and $\mathcal{D}^P$, and the loss $\mathcal{L}_{\mathcal{D}^*}$ of the generator $\mathcal{G}$, respectively. In Figure 3.12(c), the $\mathcal{L}_{\mathcal{D}^*}$ of AEP_{MTRM} shows similar tendency to the $\mathcal{L}_{\mathcal{D}^*}$ of AEP_{STCM} when the number of training step is below than 10K. The generator losses in Figure 3.12(d) also represent that it is necessary to take more time for achieving a better solution in using random-matching manners.

These results can be thought the random-matching manner provides more diverse samples, which can cover more comprehensive representation, than the constant-matching manner, even if it takes longer time in converging a solution than the constant matching manner. Consequently, the experimental results justify the benefit of the random-matching manner to improve the robustness of AEP in detecting an anomaly of events. The experimental results demonstrate the random-matching manner can help to improve the performance on the representation learning for time-series data even though it needs a little bit longer time to train models.

3.5.7 Comparison with the state-of-the-arts

We conduct the comparison between AEP and the various AED methods. The methods selected for the performance comparison are listed as follows: MDT [62], MIP-TS [28],



(a) (b) (c)

Figure 3.13: ROC curves on the frame-level evaluation and the pixel-level evaluation using UCSD-Ped1 and UCSD-Ped2 datasets. (a) and (b) shows the frame-level and pixel-level abnormal event detection (AED) ROC curves on UCSD-Ped1 dataset. (c) illustrates the frame-level AED ROC curve. The X-axis denotes the false positive rate (FPR), and the Y-axis is defined as the true positive rate (TPR).

MPPCA [21], SFM [20], SAAFM [104], Sparse representation [26], Lu *et al.*,[98], AMDN [35], Sabokrou *et al.*,[110], Hasan *et al.*,[37], ST-Autencoder [36], Two-stream 3D-ConvNet [34], Dutta and Banerjee [114], FRCN [115], WTA+SVM [147], DeepGMM [94], NNC [116], Liu *et al.*,[125], Liu *et al.*,[122], Ionescu *et al.*,[148], Ionescu *et al.*,[117], sRNN [118], Plug-and-Play CNN [131], $\text{GAN}_{/gen}$ [124], MLAD [150], Adversarial Discriminator [132], Nguyen *et al.*,[126], DeepOC [151], BMAN [120], ISTL [127], Yan *et al.*,[142], VBHMMGD [152], and Chu *et al.*,[153]. We compare AEP with either the conventional approaches using hand-crafted features [62, 28, 21, 20, 104, 26, 98] and the recently proposed methods based on deep learning such as CNNs, recurrent neural networks (RNNs), or GANs [35, 110, 37, 36, 34, 115, 94, 131, 118, 124, 132, 126, 151]. For efficient experiment, the comparison with above methods is carried out based on AEP_{MTRM}.

UCSD-Ped dataset. UCSD-Ped1 dataset is exploited for both the frame-level evaluation and the pixel-level evaluation, and UCSD-Ped2 dataset is only used for the frame-level evaluation. Figure 3.13(a) and Figure 3.13(b) contains the ROC curves for the frame-level

evaluation and the pixel-level evaluation on UCSD-Ped1 dataset respectively. Figure 3.13(c) shows the ROC curves for the frame-level evaluation on UCSD-Ped2 dataset. In comparison with the other methods based on ROC curves, AEP shows better results than others. In the experiments using UCSD-Ped1 dataset, as shown in Figure 3.13(a), The ROC curve of AEP_{MTRM} has the steepest gradient when FPR is below than 0.1. In Figure 3.13(b), the ROC curve of AEP_{MTRM} also has a rapid gradient which is able to compare with the existing state-of-the-art methods [124, 131, 147] with an FPR in the range of 0 to 0.05, and it produces the superior ROC curve when FRP is higher than 0.05 approximately. However, as shown in Figure 3.13(c), AEP_{MTRM} doesn't always show good performance. In the frame-level evaluation on UCSD-Ped2 dataset, The ROC curve of AEP_{MTRM} produces a lower gradient than WTA+SVM [147] in the FRP interval between 0 to 0.09, even though the curve of AEP_{MTRM} shows higher position than the curve of WTA+SVM [147] in the remaining interval.

The quantitative results using AUC and EER demonstrate the superiority of AEP on AED. Table 3.6 contains the AUCs and EERs on UCSD-Ped dataset and Avenue dataset. In the experiments on UCSD-Ped1 dataset, AEP_{MTRM} achieves AUC of 97.92 and EER of 6.07 from the frame-level evaluation and AUC of 74.83 and EER of 31.06from the pixel-level evaluation. These figures surpass the previous state-of-the-art performances which are achieved by $\text{GAN}_{/gen}$ [124] and MLAD [150]. $\text{GAN}_{/gen}$ produces AUC of 97.4 and EER of 8.0 on the frame-level evaluation, and MLAD achieves AUC of 70.30 and EER of 35.00 on the pixel-level evaluation. In the frame-level evaluation on UCSD-Ped2 dataset, the best performance is achieved by Ionescu *et al.*,[117]. Ionescu *et al.*,produces AUC of 97.8, and it is higher than the AUC of 96.91 achieved by AEP_{MTRM}. The EER of AEP_{MTRM} shows the

best performance among the comparison targets. Consequently, AEP could not guarantee outstanding performance for all experiments using UCSD-Ped dataset. However, the overall experimental results show that AEP can take comparable AED performance to the existing state-of-the-art methods, and sometimes it can outperform than the others.

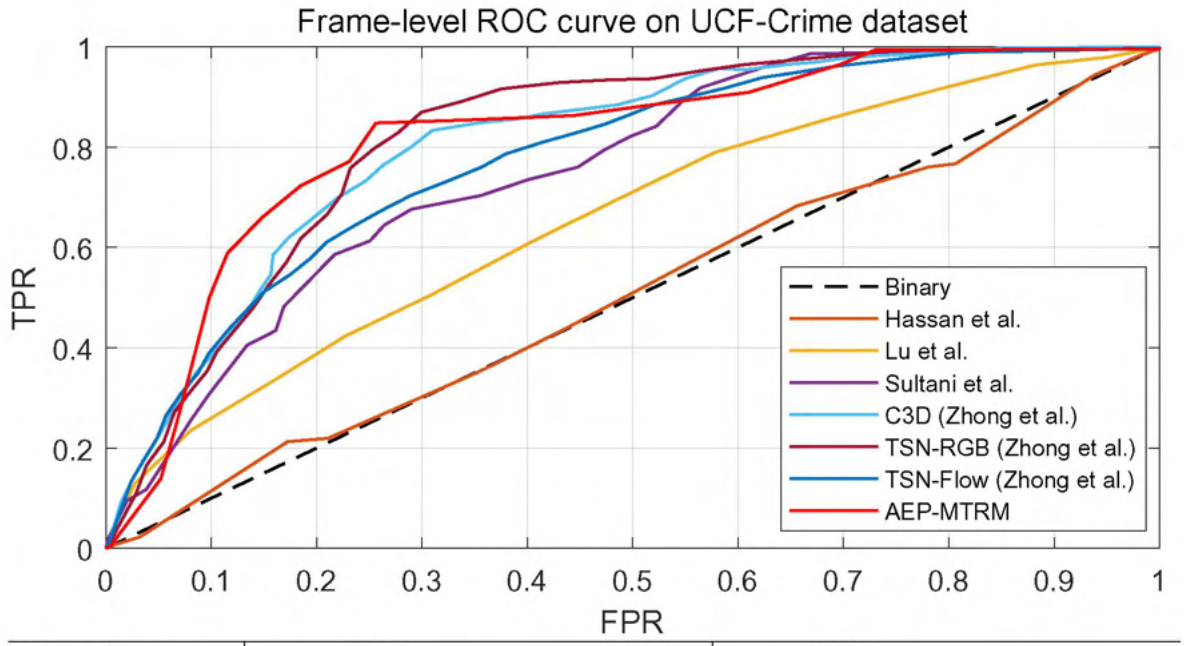
Avenue dataset. In the experiments using Avenue dataset, AEP_{MTRM} is compared with the listed methods as follows: Lu *et al.*,[98], Hasan *et al.*,[37], FRCN [115], NNC [116], Liu *et al.*,[125], Liu *et al.*,[122], Ionescu *et al.*,[148], Ionescu *et al.*,[117], sRNN [118], Wang *et al.*,[149], Nguyen *et al.*,[126], DeepOC [151], BMAN [120], ISTL [127], Yan *et al.*,[142], and Chu *et al.*,[153]. The performance of each method has been referred from their studies. In this experiments, AEP_{MTRM} archives AUC of 90.2 and EER of 10.07. These figures are lower than the state-of-the-art performance, which is AUC of 90.4 , achieved by Ionescu *et al.*,[117]. However, these figures get 2-top among the experimental results for the experiments on Avenue dataset. In piexel-level evaluation, AEP_{MTRM} achieves AUC of 94.91, and the best performance is produced by [148].

Although AEP_{MTRM} could not surpass the current state-of-the-art methods on this dataset, the performance gap between AE_{MTRM} and the existing state-of-the-art methods is below than 1%. The difference of AUC is 0.2 on the frame-level evaluation, and it is 0.49% on the pixel-level evaluation. Notably, Ionescu *et al.*,[117] exploit an object detection approach for cropping particular objects to generate training samples, and their approach is trained as a supervised learning manner requiring abnormal event samples. These components can be regarded as advantages which can affect to AED performance. As these methodological differences, the performance gap between AEP and the current state-of-the-art method may

be reasonable.

UCF-Crime dataset. The experimental results on UCF-Crime dataset show the proposed method can provide comparable performance to the state-of-the-art performance. The performance of AEP_{MTRM} is compared with the listed results on Sultani *et al.*,[99] and Zhong *et al.*,[113]. Figure 3.14 shows the ROC curves and the table contains AUCs and EERs on UCF-Crime dataset. AEP_{MTRM} achieves 81.84 of AUC and 23.04 of EER. As reported by Zhong *et al.*,, AED based on Temporal Segment Network using RGB images (TSN-RGB) achieves 82.12 of AUC, and it is the state-of-the-art performance on this dataset.

However, TSN-RGB employs BN-Inception [154] pre-trained by Kinetics-400 dataset [34] as the backbone. The network structure used to TSN-RGB is a great deeper than the network structure used in our works. In addition to the depth of the network model, the scale of the dataset used to train TSN-RGB, also much larger than the dataset exploited to train AEP_{MTRM}. Kinetics-400 dataset is composed of 300,000 video clips classified as 400 human action classes, so that it can provide more diverse event samples which can help to improve the feature representation performance of networks. Consequently, these differences about network structure and dataset can provide great advantages for AED using TSN-RGB. Therefore, it is unfair to simply compare the AUC figures. Even though TSN-RGB achieves the state-of-the-art performance on this dataset, AEP_{MTRM} can be thought that it is partially better than TSN-RGB. As shown in the ROC graph in Figure 3.14, the ROC curve of AEP_{MTRM} takes a higher position than the others in the FPR range between 0.05 to 0.27. AEP_{MTRM} achieves 23.04 of EER, and it is better than TSN-RGB's one. TSN-RGB achieves 23.54 of EER. This trend can be interpreted that AEP_{MTRM} can provide more sensitive discrimination



Method	AUC	EER
Binary Classifier	50.00	50.00
Hasan *et al.*,[37]	50.40	49.54
Lu *et al.*,[98]	65.51	39.63
Sultani *et al.*,[99]	75.41	31.20
GODS [155]	70.46	-
C3D [113]	81.08	25.29
TSN^{RGB} [113]	**82.12**	23.54
TSN^{OF} [113]	78.08	29.36
AEP_{MTRM}	81.84	**23.04**

Figure 3.14: ROC curves on the frame-level AED and the corresponding quantitative evaluation on UCF-Crime dataset. AEP trained with the multi-target random matching is compared with the methods listed on Sultani *et al.*,, and the bolded figures show the best performance among the list methods. '-' indicate the result is not provided.

power in lower threshold than other methods.

3.5.8 Analysis

As shown in the performance comparison with the existing state-of-the-art methods, we compared with various approaches including the conventional approaches based on hand-crafted features and recently proposed methods using deep learning. Particularly, among

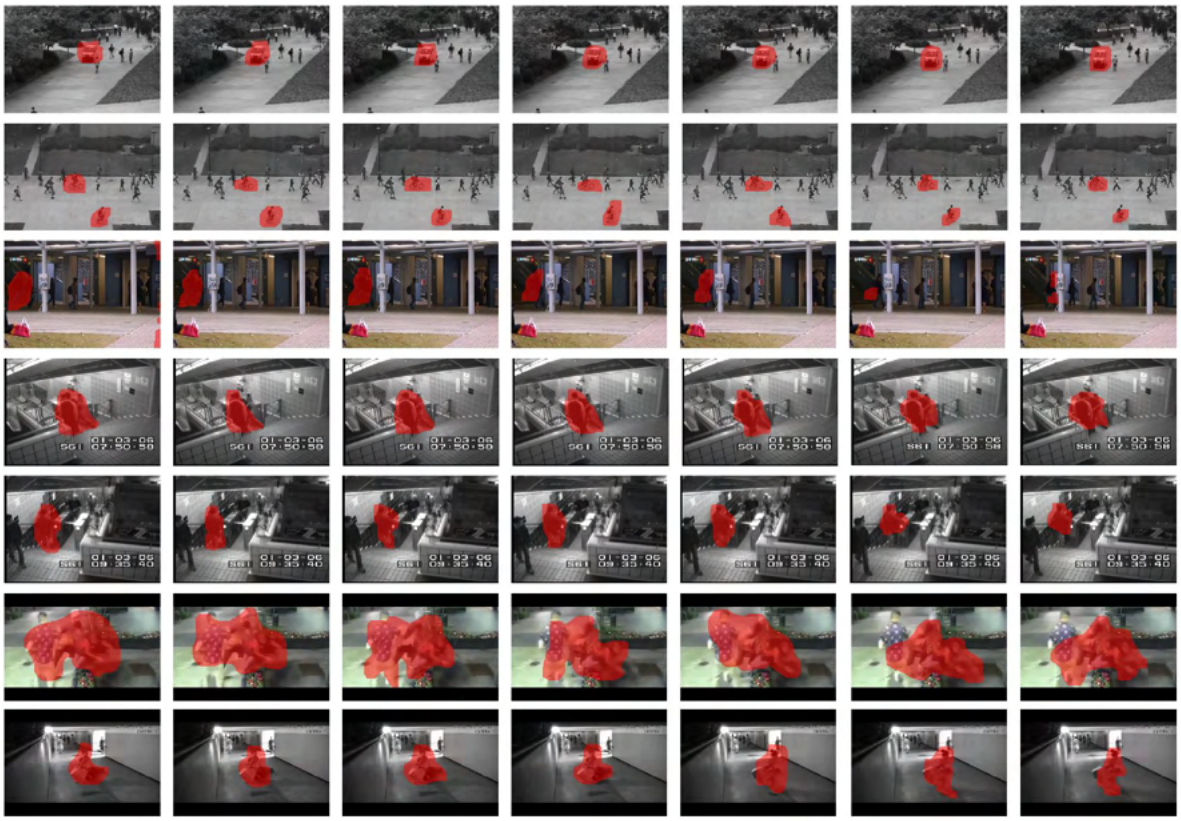


Figure 3.15: The localization results of abnormal event detection based on AEP on of UCSD pedestrian dataset, CUHK-Avenue dataset, Subway dataset, and UCF-Crime dataset. From top to bottom, the results are produced from UCSD-Ped1 dataset, UCSD-Ped2 dataset, CUHK Avenue dataset, the entrance video and the exit video on the subway dataset, and UCF-Crime dataset.

the deep learning-based methods, some studies [124, 125] show a similar approach to AED. GAN$_{/gen}$ [124] and Adversarial discriminator [130] employ a generative adversarial network for AED, and Liu *et al.*,[125] address AED as a future frame prediction problem. Additionally, various studies exploits an adversarial learning as a kind of complementary function or transform AED setting to other problem settings such as classification [116, 117, 122] or prediction [125] problems, to improve the robustness of their AED methods. Not only comparison with the approaches exploiting GANs, but also we have compared AEP with the approaches which deal with AED by transferring to other problem domains such as classification setting [116, 117, 122].

Notably, AEP presents superior AED performance than the others employing GANs [124, 125, 132]. Additionally, AEP's AED performance surpasses Liu *et al.*,[125] and Liu *et al.*,[122], which deal with AED as the future frame prediction and use optical flow and prepared abnormal event samples in the training step. These results demonstrate that the proposed adversarial learning for events' past and future, can improve the AED performance without auxiliary information such as optical flow and explicit abnormal event samples in the training step, by forcing AEP to learn the mapping function to predict event's future and preventing that the function learns a wrong correlation for events' past.

In some experiments, AEP does not show the best performance for the following reason. In training AEPs, AEP does not utilize pre-processing procedures, e.g. object detection, which can remove noise information such as background or frames which contains nothing for events. Some part of the training samples included in Avenue dataset does not contain any events and even some frames have some noise such as motion blurring.

These issues can affect to APE's mapping function because it can lead AEP to poorly optimized solution in the training step. To handle this issue, object-centric approaches [117, 113] are recently proposed, and Ionescu *et al.*,[117] shows better results than AEP in the experiments on UCSD-Ped2 dataset and Avenue dataset. However, even though AEP could not achieve the best performance for some part on datasets used for evaluation, the overall experimental results demonstrate that AEP can outperform the existing state-of-the-art methods.

3.6 Conclusion and Discussion

We propose AEP for event anomaly detection and localization. AEP derives the mapping function in order to model the correlation between events' current and future using the adversarial learning, which can help to improve prediction performance on AEP. The experimental results demonstrate that AEP trained by the proposed adversarial learning approach can provide better performance than the ones produced by existing state-of-the-art methods for AED.

Although AEP achieves the state-of-the-art performance on AED, there are some drawbacks which should have to consider in the future. First, a large-scale dataset for normal events is necessary to establish a well-generalized modal to cover the various scenarios even though AEP is learned by the generative approach based on unsupervised learning. This issue, however, is an inherent problem for almost all existing methods based on deep neural networks for visual recognitions. Second, AEP is based on 3D-CNNs so that it requires a high computational resource to operate it, and the computational cost to localize the results on AED is exponential since the localization is carried out using the siding-window technique.

These drawbacks would be taken into account for our future work. Primarily, we would like to study about optimization methods which can improve the representation learning performance on AED, even if a model cannot utilize a large-scale and well-categorized datasets. In second, we would like to develop the knowledge distillation method to downsize the network applied in AED in order to reduce the computational cost.

Methods	Ped1 (Frame)		Ped1 (Pixel)		Ped2 (Frame)		Avenue (Frame)		Avenue (Pixel)	
	AUC	EER	AUC	EER	AUC	EER	AUC	EER	AUC	EER
MDT [62]	81.4	25	44.1	58	82.9	25	-	-	-	-
MIP-TS [28]	-	-	64.9	41.3	-	-	-	-	-	-
MPPCA [21]	59	40	20.5	81	69.3	30	-	-	-	-
MPPCA+SFM [21]	66.9	32	21.5	72	61.5	35	-	-	-	-
SFM [20]	67.5	31	19.27	79	55.6	42	-	-	-	-
SAAFM [104]	77.6	29	-	-	-	-	-	-	-	-
Sparse representation [26]	89.5	19	50.2	53	-	-	-	-	-	-
Lu *et al.*,[98]	91.8	15	63.8	59.1	-	80.9	27.5	-	92.9	-
AMDN [35]	92.1	16	67.2	40.1	90.8	17	-	-	-	-
Sabokrou *et al.*,[110]	-	-	-	-	82.4	19	-	-	-	-
Hasan *et al.*,[37]	81.0	27.9	-	-	90.0	21.7	76.9	34.0	-	-
ST-Autencoder [36]	89.9	12.5	-	-	87.4	12	-	-	-	-
Two-stream 3D-ConvNet [34]	86	22	64.4	45	83.2	24	-	-	-	-
Dutta and Banerjee [114]	-	19.8	-	-	-	22.3	-	-	-	-
FRCN [115]	-	-	-	-	92.2	13.9	89.8	16.7	-	-
WTA+SVM [147]	81.3	27.9	56	46.8	96.6	8.9	-	-	-	-
DeepGMM [94]	92.5	15.1	69.9	64.9	-	-	-	-	-	-
NNC [116]	-	-	-	-	-	-	88.9	-	94.1	-
Liu *et al.*,[125]	83.1	-	-	-	95.4	-	84.9	-	-	-
Liu *et al.*,[122]	71.8	-	-	-	92.21	-	84.4	-	-	-
Ionescu *et al.*,[148]	68.5	-	52.4	-	82.2	-	82.6	-	**95.4**	-
Ionescu *et al.*,[117]	-	-	-	-	**97.8**	-	**90.4**	-	-	-
sRNN [118]	-	-	-	-	92.21	-	81.71	-	-	-
Plug-and-Play CNN [131]	95.7	8.0	64.5	40.8	88.4	18.0	-	-	-	-
$GAN_{/gen}$ [124]	97.4	8.0	70.30	35.02	93.5	14	-	-	-	-
Wang *et al.*,[149]	90.05	13.5	-	-	89.9	11.5	90.3	15.5	-	-
Zhong *et al.*,[113]	-	-	-	-	92.8	-	-	-	-	-
MLAD [150]	82.34	23.5	70.30	35.00	93.5	14	-	-	-	-
Adversarial Discriminator [132]	96.8	7.0	70.8	34.00	95.5	11	-	-	-	-
Nguyen *et al.*,[126]	-	-	-	-	96.2	-	86.9	-	-	-
DeepOC [151]	83.5	23.4	63.1	-	96.9	8.8	86.6	18.5	-	-
BMAN [120]	-	-	-	-	96.6	-	90.0	-	-	-
ISTL [127]	75.2	29.8	-	-	91.8	8.9	76.8	29.2	-	-
Yan *et al.*,[142]	75.0	32.4	67.6	-	91.0	15.5	79.6	27.5	90.6	-
VBHMMGD [152]	-	29.0	-	-	-	13.8	-	-	-	-
Chu *et al.*,[153]	90.9	16.2	-	-	90.2	17.3	82.1	-	93.7	-
AEP_{MTRM}	**97.92**	**6.07**	**74.83**	**31.06**	97.31	**7.52**	90.2	**10.07**	94.91	**10.2**

Table 3.6: Quantitative performance comparison of the AED methods using UCSD-Ped dataset and Avenue dataset. "-" means the results are not provided. The **bolded** figures indicate that the best performance among them.

Chapter 4

Drowsiness Detection for Intelligent Vehicle

4.1 Driver Drowsiness Detection

Driver drowsiness detection is one of the essential functions in the advanced driver assistant systems (ADAS) for preventing fatal accidents from the people on a road. Many drivers and pedestrians are killed or significantly injured by drowsy driving. The report of the National Sleep Foundation's Sleep in America poll presents 60% of Americans have an experience of drowsiness driving, and 37% have experienced falling asleep while driving in the recent one year. According to the report of the national highway traffic safety administration in the USA, the driver fatigue is closely related to the 100,000 of car crashes reported by polices. By this report, this car crashes made 1,550 deaths, 71,000 injuries, and 12.5 billion in monetary losses [156]. Also, the car crash by the driver drowsiness is not unique to drivers in the USA, drowsiness contributes to as many as 7% of crashes in the United Kingdom and 3.9% of crashes in Norway[157, 158]. The majority of drowsiness-related car accidents, approximately 80%, might be classified as individual vehicle run off road crashes, where a driver lost the controlling their vehicle and eventually departed their lane or smashed into the rear of the car ahead [159]. These figures may be the tip of the iceberg because of not only it is hard to attribute the cause of crashes to drowsiness but also the criteria for recognizing drowsiness differ depending on the driver [156]. There is no Breathalyzer equivalent for

drowsiness. Therefore, in order to prevent these losses of life and property, it is an important challenge to develop a driver drowsiness detection method.

The approaches for driver drowsiness detection could be classified based on their target domain to analysis. One approach is to directly analyze the driver's behaviour to identify changes in driver behaviour. This approach analyzes facial elements such as eye and mouth using visual sensors [160, 161, 162, 163, 164, 165, 166], or detects particuar patterns in electrophysiological signals occurring when a driver is falling asleep [167, 168, 169, 170]. Other approaches indirectly infer a driver's state through analysis of signals extracted from the steering system [171, 172, 173, 174, 175].

The most commonly applied and theoretically rigorous approach involves the analysis of electrical bio-signals e.g., electroencephalogram (EEG) or facial elements such as eye based on percent eye-closure over a fixed time window (PERCLOS) [176]. Dinges et al. had verified that the approach using PERCLOS had over than 90% accuracy in recognizing degraded performance during a vigilance task. This figure demonstrated that the PERCLOS was more reliable across drivers than EEG, blinks, and head position in the study [176]. Khushaba et al. proposed the driver drowsiness detection method which employs fuzzy mutual-information-based wavelet packet transform model for extracting drowsiness-related information from a set of EEG, electrooculogram (EOG), and electrocardiogram (ECG) signals [167]. Papadelis et al. developed drowsiness monitoring system using onboard electrophysiological recording systems [170]. Aforementioned methods identify the change of patterns of signals such as brain activity or heartbeat to measure the strength of fatigue of drivers. These signals reflect brain electrical activity and can provide more discriminative information than other features

in analyzing the driver's conditions. For these reasons, the methods using biomedical signals captured from drivers had provided relatively higher accurate detection results than other methods based on visual analysis or measuring the steering signals. Nevertheless, the main disadvantage of these methods is that the sensing equipment for the physiological signals such as EEG, ECG, and EOG, must be attached to the driver's body. The attachment of those sensors could cause inconvenience to drivers when they are driving. Additionally, the high price of sensors is one reason that they can not be used in a practical drowsiness detection system.

In addition to the methods of directly recognizing the drivers' condition through the analysis of biomedical signals, the approaches based on visual analysis of facial elements generally employ computer vision techniques such as object detection and tracking to find the interesting objects such as eye or mouth, on the image containing the driver's face [160, 161, 162, 163, 164, 165, 166]. Garcia et al. proposed a system which consist of three steps [160]. Their system initially detects and tracks face and eye, and then to stabilize the performance of analyzing the status of the eye in various illumination conditions, the system conducts image filtering. This system evaluates the closure status of the eye using PERCLOS measurement. Mbouna et al. provided the analysis method for a visual feature to understand the closure state and head pose. The proposed method monitors a driver using a single camera without any source of light [161]. Wang et al. presented a solution for the situation that driver is wearing glasses by combining two analysis methods for the status of eye and mouth [162]. The method proposed by Dwivedi et al. extracts features using a convolutional neural network and detects eye blinking, eye closure, and yawning [177]. Generally, these methods assume

that facial expressions of extremely tired drivers, such as eye blinking, yawning, and eye and head moving, are different from facial expressions represented when drivers are not tired. These approaches classify the driver's condition as whether he/she is asleep or not, using the hand-crafted features such as the histogram of gradient (HoG) [60] and Haar-like features [178]. To extract these facial feature information, visual sensors like an RGB camera or an active infrared sensor should be installed on the vehicle dashboard, sun visor, or overhead console for taking face images of drivers. However, despite the convenience of installation, the methods based on video analysis using visual sensors solely, provide unstable detect results in many situations. For example, general cameras cannot capture clear images at night without illumination system. The development of the drowsiness detection method using visual analysis, invariant to the light condition is still an open question.

The limitations of the above-mentioned approaches have led researchers to attend to the signals from a steering system such as the deflection of the top of the wheel from the zero point [179]. These signals are similar to electrical bio-signals in that they require significant pre-processing and transformation before they become viable input measures [180]. Sayed and Eskandarian proposed a steering-wheel angle based method that filtered raw information for steering angle for the elimination of road curvature events, and then discretized into binary signals to represent steering patterns [180]. This method detected the drowsiness of drivers with nearly 90% accuracy. Similarly, Krajewski et al. presented an approach to process raw steering-wheel angle data into features represented by the signal in the time and frequency domains [181]. Ersal et al. presented an approach to recognition of driving behaviours [171], which is based on support vector machines (SVM) [182]. This

approach systemically assists determination of whether a driver is asleep or not by interpreting behaviours of drivers using the linear discriminative model. Takei et al. [174] estimated a driver's fatigue by analyzing steering motions with the fast Fourier transform (FFT) and Chaos characteristics. These methods judge whether a driver is falling into a drowsy state by analyzing signals such as variation of velocity, acceleration, breaking, and gear change, that are recorded from the sensors embedded in steering systems. These methods are not focused on the detection of driver drowsiness directly. They try to recognize the unstable vehicle movements that are caused by various intrinsic and extrinsic reasons from analyzing steering signals. Consequently, it can provide a more flexible system to detect unstable movements than other systems which are only focused on the detection of driver drowsiness. However, many automobile manufacturers in the world embed a particular steering system in their vehicles. In addition, these signals cannot be a clear basis to distinguish whether a driver is sleepy or not since every driver has not only a different personality but also a different driving habit.

Recently, deep learning architectures have been successfully used to solve various computer vision problems, such as image recognition [29, 183], object detection [30, 184], gesture recognition [31], image segmentation [185], and action recognition [32, 186]. In particular, the deep learning methods [32, 186] show good performance in analyzing video streams to recognize specific actions when compared with conventional methods based on hand-crafted features [38, 39]. Although various methods [38, 39, 40] to extract superior hand-crafted features have been proposed, the key to these successes is a rich and discriminative representation extracted from multi-layer nonlinear systems in the deep learning approaches [35].

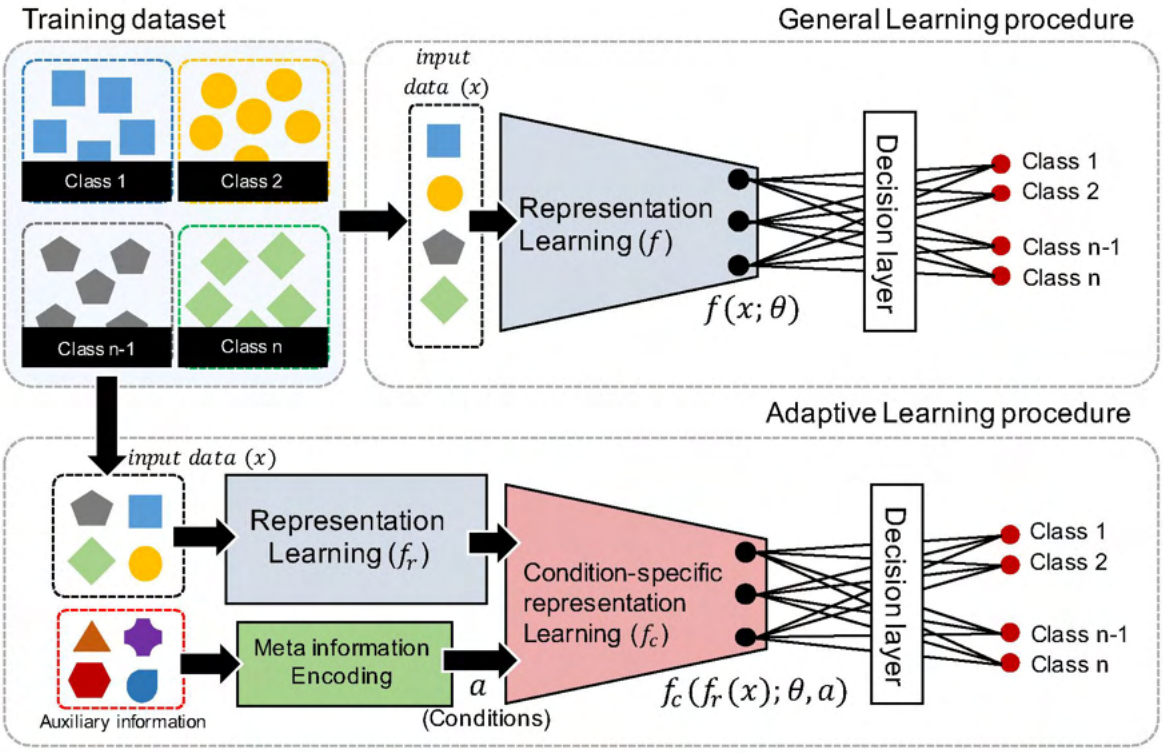


Figure 4.1: Illustrations of the processes of general represenration learning and adaptive representation learning on a classification task

We had adopted the convolutional neural network (CNN) and multi-layer fully connected neural network (a.k.a., deep neural network) to discover significant time-space features, and showed the possibility of the deep learning method for drowsiness detection in previous works [187]. In our previous works, we had proposed the driver drowsiness detection method exploiting extra scene condition prediction to improve discriminative properties of learnt representation. However, despite outperforming in drowsiness detection, the previous method had a critical drawback in generating representations. The previous method had a possibility that the method generates extremely sparse representation which cannot contain sufficient information to detect drowsiness. This work is improved and extended from our earlier work [187], and we propose an end-to-end learning framework for a novel representation called

self-reinforced representation for drowsiness detection.

The self-reinforced representation learning is a representation learning process to take the feature focused on some particular condition using auxiliary information (a.k.a., meta information). When the training dataset can be classified to several conditions, whilst the normal representation learning perform to extract generalized features from overall training data the self-reinforced representation learning can extract more specific representations reflecting given conditions. Figure 4.1 represents the comparison of processes about the normal representation learning and self-reinforced representation learning. An auxiliary information has been used to improve the performance of the deep learning model in many computer vision studies [188, 189]. Hong et al. proposed deep learning system using transferrable knowledge to the scene segmentation in training phase [188]. Zhang et al. proposed a face alignment method using the result of landmark detection as auxiliary information [189]. These methods tried to improve the performance of their solutions by learning the features biased to extra information that could help to explore useful features in their target domains. As with the methods described above, the concept of the self-reinforced representation could be possibly interpreted as a representation biased to some conditions. However, in compared to the above methods which use extra information solely in training phase as prior knowledge, the proposed framework can generate the information which can help to improve the discrimination of the learnt representation during not only the training task but also testing task. By using this paradigm, the proposed framework can immediately generate the representation which adapts to the interpreted results.

The proposed framework is composed of four models consisting of representation learning,

scene understanding, feature fusion, and drowsiness detection. The representation learning model discovers the rich and discriminative representation that can describe the motion and appearance of an object within the consecutive frames simultaneously. The scene understanding model identifies the various scene conditions that relate to driving conditions, e.g., illumination conditions and wearing glasses. The feature fusion model generates a self-reinforced representation which is biased to a specific scene condition as opposed to the general spatio-temporal representation. The proposed framework detects drivers drowsiness in various situations accurately by using this self-reinforced representation. The main contribution of this work is the representation learning framework that could be adapted to the particular scene conditions via understanding the scenes and generating the condition adaptive representation.

4.2 Self-reinforced Representation Learning Framework

4.2.1 Architectural details

The proposed framework is based on four models for representation learning, the scene understanding, the feature fusion, and the drowsiness detection. The representation learning model f_d based on 3D-DCNN is used to extract the spatio-temporal representation from an input data. The scene understanding model consists of four sub-models f_{gl}, f_h, f_m, f_e for interpreting the condition of glasses, illuminations, and movement of facial elements. The fusion model f_{fu} generates self-reinforced representation which can acclimatize the scene conditions. The detection model f_{det} determines whether a driver is sleepy or not. Figure 4.2 shows an overall architecture of the proposed framework. The brief explanation for how to

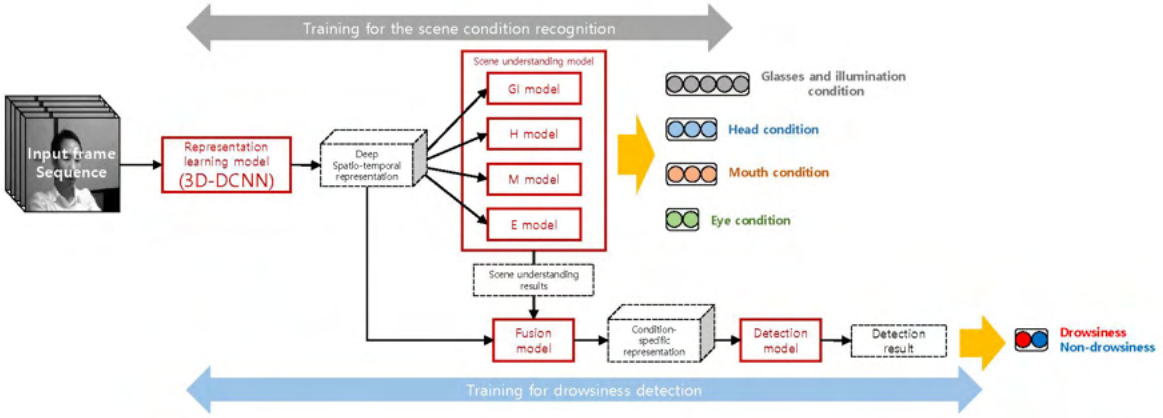


Figure 4.2: Overall architecture of the proposed framework. The red boxes with bold line denote the models, and the black boxes drawn by dotted line define extracted features or outputs of each model.

generate self-reinforced representation and detect drowsiness of drivers, using the proposed framework is as follows. Initially, the representation learning based on the 3D-DCNN extracts a feature that can describe motion and appearance from a video clip simultaneously. Secondly, the scene understanding predicts five scene conditions that associated with wearing glasses, illumination conditions, and facial elements using the spatio-temporal feature extracted from the representation learning. The scene understanding results are represented by a vector that is defined by the one-hot encoding method. The one-hot encoding is one of the encoding approaches which indicates the state of a system using the binary values. The encoding result is represented by the group of bits among which the legal combinations of values are only those with a single high (1) bit and all the others low (0) bits. Then, feature fusion learns a self-reinforced representation by agglomerating the spatio-temporal representation and the one-hot vectors. Finally, the detection model identifies a state of driver drowsiness by analyzing the self-reinforced representation. In the following, we will describe the detail of information of each model and training scheme of the proposed framework.

4.2.2 Spatio-temporal representation learning

In this section, we describe the representation learning model using 3D-DCNN for extracting the spatio-temporal representation from given mutlitple consecutive frames. The objective of the representation learning is discovering a rich and discriminative feature from inputted consecutive frames. Videos taken by the frontal facing camera in the display units of a vehicle can be variously modified depending on the various conditions of the vehicle interiors or exteriors, such as illumination conditions and an interior design of a vehicle. When drivers feel drowsiness, their facial elements make various changes, and these changes would be interpreted as either a shift in shape or change of motion. Therefore, to detect a drowsiness of drivers, we have to consider the representation which can describe spatial information (appearance) and temporal information (motion) simultaneously. It is impossible to estimate a temporal information using only a single frame since a single frame cannot contain a change according to a time sequence. When we consider these limitations observed when a input is a single frame, it is necessary to use multiple consecutive frames as an input to discover the spatial and temporal information simultaneously. In this work, we employed 3D-DCNN to discover various spatial and temporal change in given multiple consecutive frames.

Let $x \in R^{W \times H \times T}$ denotes a training video clip where W, H, and T are the width, height, and the temporal length respectively. For a given input video clip x, the representation learning based on the 3D-DCNN extract a spatio-temporal representation as

$$\boldsymbol{a} = f_d(x; \theta_d), \quad \boldsymbol{a} \in R^{W_a \times H_a \times D_a} \tag{4.1}$$

where θ_d is the parameter vector of the representation learning, and $\boldsymbol{a}$ is a learnt spatio-temporal representation. The spatio-temporal representation is defined as the activation values of the hidden units in the last convolutional layer of 3D-DCNN of the representation learning model. W_a, H_a, and D_a denote the width, height, and depth of the spatio-temporal representation. The 3D-DCNN in the representation learning is composed of six convolutional layers and two pooling layers. Figure 4.3 shows the architectural detail of the 3D-DCNN in the representation learning. To discover a spatial and temporal feature simultaneously, we employed a 3D local receptive field suggested by Tran et al. [33]. The convolutional operation based on 3D local receptive field can be defined as

$$a = \rho[\sum_{i}^{W_r}\sum_{j}^{H_r}\sum_{k}^{D_r}(v_{i,j,k}w_{i,j,k} + b)] \tag{4.2}$$

where a is an activation value of the hidden unit, and v, w, and b are the input value, the weight, and bias respectively. W_r, H_r, and D_r denote the width, the height, and the depth of 3D local receptive field, and ρ is an activation function for the convolution layer. We adopt the Rectified Linear Units (ReLUs) [5] for the proposed 3D-DCNN. While the ordinary 2D structure of the kernel (local receptive field) in 2D convolution layers can extract spatial information only, the 3D structure of the kernel in 3D convolution layer allows to us capturing the spatial and temporal features simultaneously. The extracted representations which contain spatial and temporal features convey to the scene understanding model and feature fusion model to identify the various scene conditions and generate the self-reinforced representation.

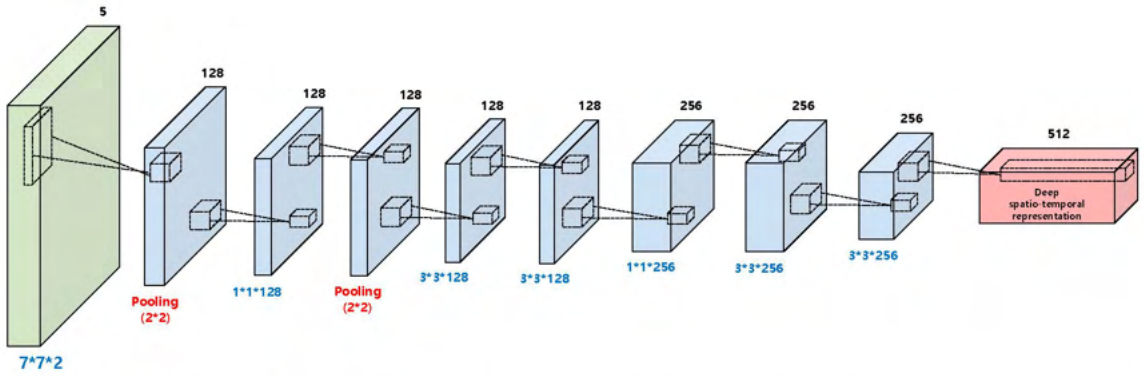


Figure 4.3: Illustration of the 3D-DCNN in representation learning module. The green box and red box denote an input data and extracted spatio-temporal representation respectively, and the blue boxes represent convolution layers and pooling layers. Numbers located in the upside of the boxes represent the depth of each layer, and numbers below the boxes illustrate the dimensionality and structural detail of the kernel in each convolutional layer.

4.2.3 Scene understanding

The goal of the scene understanding is interpreting of the scenes with drivers, and understanding the various condition of drivers that can be categorized by the physiological and environmental conditions such as movement of facial elements, wearing glasses, and a difference between a day and night. These interpreted information help to train the framework for adapting the learnt representation to the various scene conditions. We hypothesize that each video clip is associated with the scene conditions and a driver drowsiness status. These are represented by either ground-truth (in training phase) or prediction results (in the inference phase).

In this work, the scene condition contains the three categories of the facial elements and one category for the status of glasses and illumination: 1) conditions of glasses and illumination $\mathcal{L}_{gl}$, 2) head $\mathcal{L}_h$, 3) mouth $\mathcal{L}_m$, and 4) eye $\mathcal{L}_e$. We define states of facial elements and the conditions for glasses wearing and illumination using a one-hot vector. The detailed

explanation for the annotation of each scene condition is described in Table 4.2.3. We adopt a fully connected neural network since there is a possibility that given spatiotemporal representations have complex distributions which can not be modelled by a linear kernel. The predictions of conditions using the scene understanding model are written by

$$\begin{aligned}
\hat{\mathcal{L}_{gl}} &= f_{gl}(\boldsymbol{a}; \theta_{gl}), \quad \mathcal{L}_{gl} \in R^{L_{gl} \times 1} \\
\hat{\mathcal{L}_{h}} &= f_{h}(\boldsymbol{a}; \theta_{h}), \quad \mathcal{L}_{h} \in R^{L_{h} \times 1} \\
\hat{\mathcal{L}_{m}} &= f_{m}(\boldsymbol{a}; \theta_{m}), \quad \mathcal{L}_{m} \in R^{L_{m} \times 1} \\
\hat{\mathcal{L}_{e}} &= f_{e}(\boldsymbol{a}; \theta_{e}), \quad \mathcal{L}_{e} \in R^{L_{e} \times 1}
\end{aligned} \tag{4.3}$$

where $\hat{\mathcal{L}} \in \{\hat{\mathcal{L}_{gl}}, \hat{\mathcal{L}_{h}}, \hat{\mathcal{L}_{m}}, \hat{\mathcal{L}_{e}}\}$ are predicted scene conditions associated to input data x, and $L \in \{L_{gl}, L_{h}, L_{m}, L_{e}\}$ are dimensions of each annotation for the condition containing glasses and illumination, head, mouth, and eye. $\theta \in \{\theta_{gl}, \theta_{h}, \theta_{m}, \theta_{e}\}$ are the parameters of the each model that defined by the fully connected network in the scene understanding model. Each model is composed of two hidden layers and a corresponding output layer. The aforementioned models are represented as

$$o = f_{o}\{f_{h2}[f_{h1}(aW_{h1} + b_{h1})W_{h2} + b_{h2}]W_{o} + b_{o}\} \tag{4.4}$$

where f_{h1}, f_{h2}, and f_{o} are activation functions of the first and second hidden layers and an output layer respectively. a is reshaped a spatio-temporal representation which is extracted from the representation learning model based on 3D-DCNN. W_{h1}, W_{h2}, and W_{o} are weight parameters of two hidden layers and the output layer. b_{h1}, b_{h2}, and b_{o} are the bias parameters

of each layer. The learning procedure of each sub-model in the scene understanding is similar to the back propagation algorithm [190]. Each sub-model estimates a condition that corresponding to the given spatio-temporal representations $\boldsymbol{a}$, then computes the difference between the predicted conditions and annotations to train the parameters of the network of the sub-model. The dimensionalities of the outputs for each scene understanding model correspond to their target domain to predict. For example, the dimensonality of the output o of the scene understanding model for glasss and illumination conditions is five, because of the model is designed to identify the conditions defined as five classes. For a given spatio-temporal representation as input, the scene understanding model is trained to optimize the objective function defined as follows

$$\begin{aligned} E_{su}(\hat{\mathcal{L}}, \mathcal{L}; \theta) = \min_{\theta_d, \theta_{gl}, \theta_h, \theta_m, \theta_e} \beta \sum_i [E_{gl}(\mathcal{L}_{gl}, \hat{\mathcal{L}}_{gl}) \\ + E_h(\mathcal{L}_h, \hat{\mathcal{L}}_h) + E_m(\mathcal{L}_m, \hat{\mathcal{L}}_m) + E_e(\mathcal{L}_e, \hat{\mathcal{L}}_e)]. \end{aligned} \tag{4.5}$$

where $\mathcal{L} \in \{\mathcal{L}_{gl}, \mathcal{L}_h, \mathcal{L}_m, \mathcal{L}_e\}$ denote annotations of input data, and E_{gl}, E_h, E_m, and E_e denote loss functions defined by the softmax cross-entropy loss between the annotation and predicted results. β is a hyper-parameter for regularization of the summation of values of error functions. The details of training and inference tasks are given in Section 4.3. The spatio-temporal representation and the outputs of the scene understanding model are then combined to produce the self-reinforced representation explained in the following subsections.

Table 4.1: Annotations for the sub-models in the scene understanding and its status.

Scene condition	Category	One-hot vector	Condition
Glasses and illumination conditions	1	10000	Day bare face
	2	01000	Day glasses
	3	00100	Night glasses
	4	00010	Night bare face
	5	00001	Day sunglasses
Head condition	1	100	Normal status
	2	010	Looking at both sides
	3	001	Nodding
Mouth condition	1	100	Normal status
	2	010	Talking and laughing
	3	001	Yawning
Eye condition	1	10	Sleepiness eye
	2	01	Normal status

4.2.4 Feature fusion

The objective of the model for feature fusion is to learn a set of self-reinforced representations from the given spatio-temporal representation $\boldsymbol{\alpha}$ and its associated scene condition annotations $\hat{\mathcal{L}} \in \{\hat{\mathcal{L}_{gl}}.\hat{\mathcal{L}_h}.\hat{\mathcal{L}_m}.\hat{\mathcal{L}_e}\}$. Given the spatio-temporal representation extracted from 3D-DCNN $\boldsymbol{\alpha} \in R^{W_\alpha \times H_\alpha \times D_\alpha}$ and its associated and predicted scene conditions $\hat{\mathcal{L}}$, the fusion model discovers a set of self-reinforced representation β. The self-reinforced feature vector β is generated by using the multiplicative interaction approach proposed by Memisevic et al., [191]. Hong et al. observed that the high-order dependency between relevant features can be captured by using element-wise multiplication interaction between the feature maps [192]. To train the proposed framework that generates the combined representation which needs joint learning between the multiple resources, we refer to the training procedure proposed by Hong

et al., [192]. The fusion model is defined as follows

$$\beta = f_{fu}(\boldsymbol{\alpha}, \mathcal{L}; \theta_{fu}) \tag{4.6}$$

$$\begin{aligned} \beta = W_{fu}(W_{fea}\boldsymbol{\alpha} \otimes W_{gl}\mathcal{L}_{gl} \otimes W_h\mathcal{L}_h \\ \otimes W_m\mathcal{L}_m \otimes W_e\mathcal{L}_e) + b_{fu}. \end{aligned} \tag{4.7}$$

where β denotes the unnormalized self-reinforced representation, $b_{fu} \in R^{d\times 1}$ is the bias of the fusion model, and $\otimes$ denotes element-wise multiplication. The weights are given by $W_{fu} \in R^{M\times d}$, $W_{fea} \in R^{d\times W_\alpha H_\alpha D_\alpha}$, and W_{gl}, W_h, W_m, and W_e are defined as the specific sizes based on the dimensional scale of each associated annotation. The variables M and d denote the number of hidden units in the fusion model. This 5-way tensor product can capture the correlation between the input domains containing the spatio-temporal representation and the scene conditions.

However, the element-wise multiplication with the spatio-temporal representation and the outputs of the scene understanding empirically computes values that are close to zero. These computed values can influence not only the result of the fusion model but also computational procedure when the multiplication results exceeded the range that can be represented by computation machine. We adopted a normalization scheme to prevent values close to zero for avoiding the computational errors and finding high-order dependency between the spatio-temporal representation and the identified scene conditions. To prevent computational error and to pay attention to only a scene condition, we normalize β to v using the softmax function

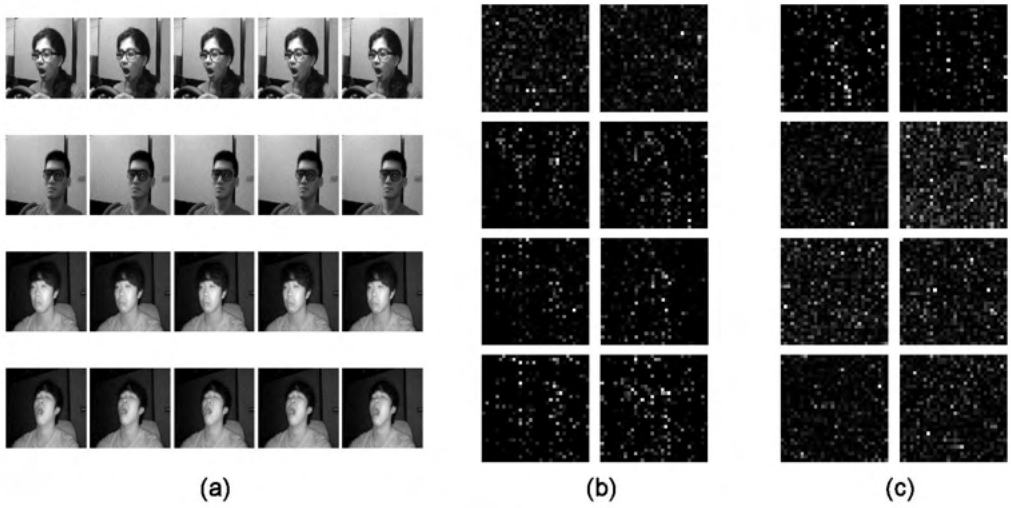


Figure 4.4: Illustration of the deep spatio-temporal representation and condition-adaptive representation according to input data. (a) Input frames, (b) Deep spatio-temporal representation, and (c) denotes condition-adaptive representation obtained by the fusion model f_{fu}. Two images in (b) and (c) represents the visualization of activation results of hidden units in representation learning and feature fusion modules. The proposed condition-adaptive representation learning framework adaptively discover the conditional feature in an input volumes depending on the result of the scene understanding model.

in [185, 193]. The normalization is formulated as follows

$$v_i = \frac{exp(\beta_i)}{\sum_j exp(\beta_j)} \tag{4.8}$$

where β_i represents i-th element of the unnormalized joint feature, and v_i is i-th element of the normalized fusion feature. Intuitively, v represents a self-reinforced representation defined over all spatio-temporal representations and the corresponding scene conditions. Figure 4.4 shows the input images, the spatio-temporal representations, and the self-reinforced representations. The self-reinforced representations are then used as an inputs to the detection model, which is explained in next section.

4.2.5 Drowsiness detection

The fusion model described in the previous subsection generates a set of self-reinforced representations v, which provide scene adaptive features containing information of facial elements and illumination of drivers. The drowsiness detection of the proposed framework using the given self-reinforced representation v in Eq. (10) is carried out via additional neural networks. As same as the scene understanding model, we put an additional fully connected deep neural network on top of the fusion model as follow:

$$o_{det} = f_{det}(v; \theta_{det}). \tag{4.9}$$

where o_{det} denotes the output of the detection model, and θ_{det} is the model parameter. The output of the fully connected network is consists of two units: non-drowsiness unit and drowsiness unit, to classify the drowsiness of a driver. To compute the likelihood of the driver drowsiness, we apply the soft-max function $\frac{e^{x_i}}{\sum_{k=1}^{2} e^{x_k}}$ which reflects the drowsiness and non-drowsiness degrees of input. Using the soft-max function, we can detect the driver drowsiness in each input. A high value of the non-drowsiness unit signifies that a driver in the input frames is likely to be awake, and a high value of the drowsiness unit signified that the driver is falling asleep. An optimization scheme for both f_{fu} and f_{det} operates under the detection objective. Our detection model is trained to minimize the detection loss using detection annotation associated with fusion feature, and representation as follows:

$$\min_{\theta_f, \theta_{det}} \sum_{i} E_{det}(o_{det}, \hat{o}_{det}) \tag{4.10}$$

where $\hat{o}_{det}$ is a ground-truth value that corresponds to each input data x, and E_{det} denotes the objective function of the detection model. We used the softmax cross-entropy function as the objective function for E_{det}. The objective function is worked to all models embedding into the proposed framework.

4.3 Training and Inference

The training of the proposed framework has two objectives including the scene understanding objective in Eq. (7) and the drowsiness detection objective in Eq. (12), and the harmony of those two objectives is essential for achieving a superb locally optimized solution. Combining Eq. (7) and (12), the overall objective function is defined by

$$\min_{\theta_d,\theta_{sc},\theta_f,\theta_D} \sum_i ((1-\lambda)E_{su}(\mathcal{L}_c, \hat{\mathcal{L}}_c) + \lambda E_{det}(o_D, \hat{o}_D)) \tag{4.11}$$

where λ is a parameter for balancing during training two modules for the scene understanding and drowsiness detection. The objective function can optimize the four modules of the proposed framework simultaneously. However, when we begin the training, we do not train the all models of the proposed framework simultaneously. The overall architecture (see Fig 2.) shows that the proposed framework is sharing the output of the representation learning model, and also denotes that the representation learning and scene understanding models can considerably influence to the other models (feature fusion and drowsiness detection). First, we train the representation learning and scene understanding models during n steps. After that, we train all models containing the feature fusion and detection models.

4.3.1 Data augmentation

The most general approach to reduce overfitting on a given training dataset is artificially enlarging the dataset using label-preserving transformations [5]. In this work, we apply the data augmentation based on horizontal transformation and image pyramid technique. This approach allows transformation of an image with very little computation so that we can make an additional dataset without huge computational load. We generate horizontally flipped images from the original images, and these original images and flipped images are transformed by using the image filtering methods based on the Gaussian filter. Figure 4.5 illustrates the procedure of the data augmentation. We conduct this by extracting training patches using various values of variations and training our proposed framework on this extended dataset. In our experiments, we used three different variations to generate additional training samples by using the image pyramid paradigm. These two types of data augmentation approaches can sufficiently increase the number of the training samples. Without this scheme, our proposed framework suffers from substantial overfitting, and it can converge to a poorly local optimized solution.

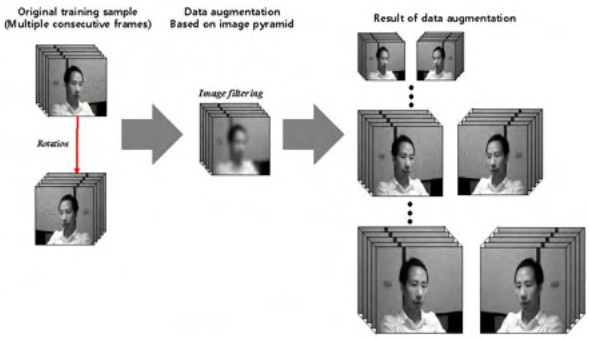


Figure 4.5: Illustration for the procedure of the data augmentation. Original training sample and the rotated sample of it generates another training samples by using the image filtering such as Gaussian filter.

To detect the drowsiness of drivers from input video clip, the proposed framework generates spatio-temporal representations using the representation learning, and then the spatio-temporal representation is used to understand scene conditions. these two pieces of information

are combined to produce the self-reinforced representation. Drowsiness is detected by using this self-reinforced representation.

4.4 Experiments

4.4.1 Benchmark dataset

Previous studies [167, 174, 175] on driver drowsiness detection attempted to recognize small cases in the private dataset which is constructed in their own experimental environment for driver drowsiness detection. Abtahi et al. provided a publicly-available dataset for yawning detection [194]. However, it is still insufficient for a comprehensive drowsy driver study. We used the NTHU Drowsy Driver Dataset (NTHU-DDD Dataset) to demonstrate an efficiency of the proposed framework for the drivers drowsiness detection. It is too difficult and dangerous to construct a dataset for detecting of driver drowsiness detection in real situations. The NTHU-DDD dataset is composed of several videos containing a driver who was sitting on a car seat and playing a racing game with driving simulator wheel and pedals. The drivers in the dataset conducted various facial expressions during video recording. The total time of the entire dataset is about 9 and a half hours.

The NTHU-DDD dataset is composed of three subsets for training, evaluation, and test, which are composed of non-redundant video files. Each subset consists of the videos which contain diverse situations for the condition for drivers that is captured using visual sensors such as a camera and an active infrared (IR) sensor. The entire dataset including training and evaluation datasets contain 36 of drivers of different ethnicities recorded with and without glasses/sunglasses under a variety of driving scenarios. The driving scenarios include normal

Figure 4.6: The example snapshots of NTHU Drowsy Driver Detection Dataset (NTHU-DDD Dataset).

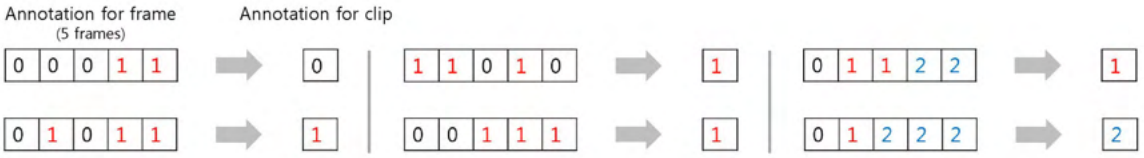


Figure 4.7: The illustration for the concept of temporal IOU.

driving, yawning, slow blink rate, falling asleep, and burst out laughing, under day and night illumination conditions. All videos contain frame-level annotation for the drowsiness condition. The video resolution is 640 $\times$ 480 in AVI format. Figure 4.6 shows example snapshots of the NTHU-DDD dataset.

The training dataset is composed of subsets that are composed of 18 subject folders. Each subject folder contains videos recorded in various driving condition. Each subset is classified into four scenarios defined as the condition of the glasses and illumination conditions (i.e., glasses, bare face, sunglasses, night glasses, night bare face). Each scenario contains four videos with different situation and corresponding annotation files. The evaluation dataset provides four subject folders and each subject contains five videos with different scenarios and corresponding annotation files. The training dataset is composed of 360 videos (722,223 frames), and the evaluation dataset contains 20 videos (173,259 frames). In this work, we

Table 4.2: Validation accuracies of the scene understanding model using the evaluation dataset in NTHU-DDD dataset.

Scenario	Glasses and illumination	Head	Mouth	Eye
Day bare face	0.99	0.99	0.98	0.89
Day glasses	0.97	0.93	0.95	0.81
Day sunglasses	0.98	0.97	0.78	0.78
Night bare face	0.99	0.95	0.97	0.82
Night glasses	0.97	0.96	0.88	0.92
Average	0.98	0.96	0.912	0.844
Total average				0.924

only used training and evaluation datasets because test dataset can not publicly accessible and the test dataset not contains annotation for performance evaluation. We used all given training data to train the proposed framework. We make a small video clip that consists of five consecutive frames, and assign an annotation about the scene conditions and drowsiness status.

Unfortunately, the given training data provides frame-level annotation, so that we employed a concept of the intersection over union (IOU) [195], in order to change the frame-level annotation to clip-level annotation. Figure 4.7 shows the concept of the temporal IOU used in our experiment. We assume that the annotation value of each clip is defined as a value occupying more than 50% among the frame-level annotations. Therefore, we defined the annotation value as the value which is observed more than three frames in each clip in our experiment. In addition, we downsample all frames using a bilinear interpolation method in Opencv library to the uniform size with width of 224 pixels and height of 224 pixels for improving an experimental and time efficiencies.

Table 4.3: Average accuracy comparison of the drowsiness detection approaches in different situations using the evaluation dataset in NTHU-DDD dataset. The **bolded values** represent the best accuracies in each scenario and the averages.

Scenario	LeNet[196]	AlexNet[5]	VGG-FaceNet[197]	LRCN[198]	FlowImageNet[198]	DDD-FFA[199]	DDD-IAA[199]	**Ours**
Day bare face	0.531	0.704	0.638	0.687	0.563	0.782	0.698	**0.796**
Day glasses	0.592	0.616	0.705	0.617	0.616	0.741	0.759	**0.781**
Day sunglasses	0.682	0.702	0.570	0.714	0.675	0.618	0.698	**0.738**
Night bare face	0.602	0.646	0.737	0.573	0.668	0.702	0.749	**0.765**
Night glasses	0.599	0.627	0.741	0.556	0.551	0.683	**0.747**	0.734
Average	0.601	0.659	0.678	0.629	0.615	0.708	0.730	**0.762**

Table 4.4: F-measures and accuracies of the drowsiness detection using for the evaluation dataset in NTHU-DDD dataset. The listed values below the drowsiness and non-drowsiness attributes represent the results of F-measures.

Scenario	Drowsiness (F)	Non-drowsiness (F)	Accuracy
Day bare face	0.809	0.784	0.796
Day glasses	0.789	0.774	0.781
Day sunglasses	0.758	0.718	0.738
Night bare face	0.753	0.777	0.765
Night glasses	0.718	0.750	0.734
Average	0.765	0.760	0.762

4.4.2 Experimental results

We demonstrate an efficiency of our framework using the evaluation set of the NTHU-DDD dataset. The evaluation dataset is composed of 5 scenarios, and each scenario contains five videos that captured various virtual driving situations. The videos in the evaluation dataset are not duplicated to the videos in the training dataset. The dataset also includes multiple annotations that are concerned with the scene conditions and drowsiness detection. We tested the performances of the scene understanding and drowsiness detection respectively.

The scene understanding module is evaluated by using validation accuracy, represented as $\frac{n}{m}$ where the numerator n is the number of the correctly classified results of each sub-model in the scene understanding model, and the denominator m denotes the total number of test samples. Table 4.2 shows the validation accuracies of the scene understanding model that is

composed of four sub-models: the glasses and illumination conditions f_{gl}, the head model f_h, mouth model f_m, and eye model f_e. The averages are computed by the formulation of the arithmetic mean so that the weights according to the number of data that classified to the same categories in the table did not consider. This measurement has been applied equally to subsequent experiments. The average of validation accuracies across to all scene conditions for sub-models is 0.924. Experimental results in Table 4.2 show that the scene understanding module in the proposed framework achieves good classification results in the classification problems of the glasses and illumination conditions and the status of a head. However, the classification result for the condition of mouth and eye is relatively lower than the other categories. The performance gaps between the sub-models in the scene understanding could be interpreted as a bias of representation learning. The understanding of the scene conditions based on our spatio-temporal representation could be influenced by the geometrical size and scale of a target object. Since the portion of each frame for an eye and mouth is relatively smaller than the portion of a frame for glasses, illumination, and head in the NTHU-DDD dataset, the learnt representation learning model would have been over-fitted to the conditions for glasses, illumination and head.

We evaluated the proposed framework quantitatively by using the F-measure. F-measure is harmonic mean of precision and detection rate, where precision and recall are defined as follows:

$$Precision = \frac{TP}{TP + FP} \tag{4.12}$$

$$Detectionrate(DR) = \frac{TP}{TP + FN} \tag{4.13}$$

$$F - measure = \frac{2 \times Precision \times DR}{Precision + DR} \tag{4.14}$$

where TP (True positive) is the number of correctly detected as drowsiness state, and FN (False negative) is the number of incorrect detection results that classified to non-drowsiness condition. FP (False positive) is the number of non-drowsiness detection result incorrectly identified to the drowsiness state, and TN (True negative) is the number of correctly classified as non-drowsiness state. The quantitative evaluation denotes an average over all videos represented as same glass and illumination categories. Table 4.4 shows the accuracy of the proposed framework for the drowsiness detection. The results show that our proposed framework achieves an average accuracy of 0.762.

Due to the lack of performance comparison using a publicly available dataset for drowsiness detection, we referred the previous method which was evaluated their performance using the NTHU-DDD dataset or implement a method based on the well-known multi-class classification algorithm for images. We compared our framework to several methods [197, 198, 199, 5, 196]. Parkhi et al. proposed a face recognition method (VGG-FaceNet) using a deep neural network [197]. The VGG-FaceNet consists of 36 convolution layers, and this network is much deeper than the 3D-DCNN used in the proposed framework. Donahue et al. provide the method based on long-term recurrent convolutional networks (LRCN) for visual recognition and description for long-term time series data [198]. We modified these methods to evaluate the performance of driver drowsiness detection. Park et al. proposed the deep drowsiness detection (DDD) network for drowsiness detection using feature-fused architecture [199]. Park et al. used two different fusion strategies to their network: independently-averaged

architecture (IAA) and feature-fused architecture(FFA). They provide the experimental results using the NTHU-DDD dataset. These methods were trained and tested with the equal procedure of the proposed framework. Additionally, we compare the results using the NTHU-DDD dataset, which is listed in Part et al.[199].

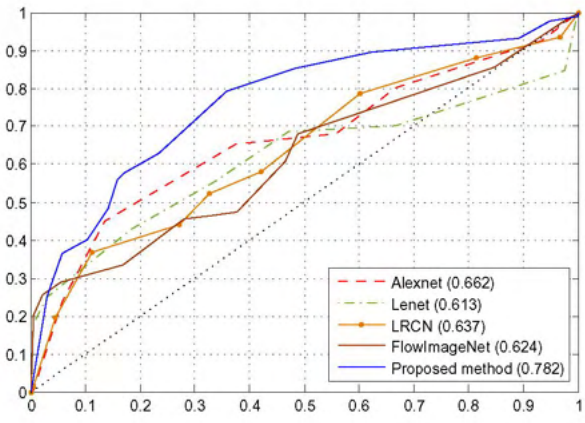


Figure 4.8: The ROCs for the driver drowsiness detection. Figures in parentheses indicate the area under curves (AUCs).

Table 4.3 shows that the comparison results of driver drowsiness detection using NTHU-DDD dataset. The experimental results show that the proposed framework outperforms other methods in most of the scenarios. Only in the night glasses scenario did the proposed method achieve a performance lower than the DDD-IAA. Additionally, the experimental results illustrate that the proposed framework achieves higher and stable performance in various scene conditions than the listed methods, even though several methods used the deeper network structure. Figure 4.8 shows the receiver operating characteristic (ROC) curves and the area under curves (AUCs), generated by the evaluation dataset predictions. The results of the ROC plots in Fig. 4.8 present that the proposed method does not take a benefit in the lower regions of the curve, where the false positive rate (FPR) is less than 0.05 approximately, but provides a definite benefit for much of the rest of the curve, over the other methods [196, 5, 197, 198].

The overall experimental results demonstrate that the proposed method can provide an

Figure 4.9: The detection results using NTHU-DDD dataset. The images of the first row show the detection results for the driver drowsiness, and the images of the second row denote the detection results of a normal condition of drivers.

accurate and effective method for the driver drowsiness detection than the other drowsiness detection method based on a visual analysis. Driver's drowsiness in the real world could appear with various variations of facial elements in diverse illumination conditions. The feature fusion helps to discover the discriminative and rich self-reinforced representation for detecting the drowsiness, and this function plays a significant role to provide high-quality drowsiness detection in various situations. Figure 4.9 shows the example snapshots of the correct detection results using NTHU-DDD dataset.

4.4.3 Computational complexity

Although the computational cost of the framework depends on the size of input images and the structure details such as the number of layers and the size of kernels in a neural network, theoretically, the computational complexity of representation learning and feature fusion models based on 3D-CNN is $O\left(\sum_{i=1}^{d} W_i H_i D_i n_i m_i k_i\right)$,

where i and d are the index of a convolutional layer and the number of convolutional layers of each model. W_i, H_i, and D_i denote the width, height, and depth of input data in each convolutional layer. n_i, m_i, and k_i denote the width, height, and depth of 3D-convolutional kernel in i-th layer. The computational complexity of the scene understanding and drowsiness detection models using two-layers neural networks is $O(N^2C)$, where N and C denote the dimensionalities of each hidden layer and target domain for objectives. We have estimated the computational complexity of the proposed framework based on the approaches of He et al., [200] and Notchenko et al., [201].

Note these computational complexities apply to both training and testing phases, however practical execution times in both phases are different since the proposed framework shows different work-flows in training and test phases. The training task consists of the three steps: 1) calculation of output, 2) computing an error, and 3) updating the parameters. Therefore, the execution time in the training task is relatively longer than the time in the testing task. Once the model training end, the execution time in testing phase is much faster because of the framework only needs to compute the output for drowsiness detection. The execution time in our experimental setting was 38.1 FPS (28.6 ms) which is almost real-time, and was obtained. We calculated this value by averaging the execution time of the proposed framework for 300 seconds, except displaying an output on a screen. The proposed framework is implemented with Google Tensorflow library. Although the training in the framework requires long times, after the model training is finished, the entire framework is able to perform in real-time with Python implementation using a Core i7, 3.4GHz PC with 16GB RAM and GTX TITAN GPU.

4.5 Conclusion and Discussion

We have proposed an self-reinforced representation learning for efficient driver drowsiness detection method which is invariant to various driving conditions containing a driving time such as day and night and a driver's appearance. To this end, we extracted the spatio-temporal representation and merged it with the vectors that represent the scene understanding results using the feature fusion method based on the tensor product approach. These problems are effectively modelled using 3D-DCNN and fully connected neural network based on recent advances in computer vision fields. The spatio-temporal representation and estimated scene conditions are merged to enhance the discriminative power for providing precise driver drowsiness detection in various driving conditions. With the feature fusion properly harnessed, the merged feature can provide more discrimination than the original spatio-temporal representation even though the original representation contains the motion and appearance information about the driving and drivers conditions. Experimental results show that the proposed framework outperforms other methods, including methods based on deep learning, in drowsiness detection accuracies.

The limitation of the proposed framework can be summarized as follows. First, although the proposed framework achieves good detection performance, it also needs a high-performance GPU computing unit that must be installed on a vehicle. It may cause high price of the vehicle and an increase in vehicle weight. Second, the proposed method needs many training samples that are labelled with the scene conditions and drowsiness state, for learning the representation that can cover various situations about drivers. Third, since the proposed framework is an off-line method, it can not guarantee to detect the drowsiness of drivers of

entirely different types that are not included in training samples.

In future works, several suggestions should be taken into account. First, we will optimize the network structure in the proposed framework for use in an embedded board or microcomputing systems to reduce the financial cost and improve the computational efficiency without performance degradation. Second, we will develop an on-line updating method in order to improve the drowsiness detection reliability of the model through continuous updating. Third, we will study a data augmentation method based on generative models to improve the performance of drowsiness detection by enlarging the scale and variety of a given dataset.

Chapter 5

Road Pavement Defect Detection

In the past few years, the performance of road defect detection has been remarkably improved thanks to advancements in various studies on computer vision and deep learning. Although large-scale and well-annotated datasets enhance the performance of detecting road defects to some extent, it is still challengeable to derive a model which can perform reliably for various road conditions in practice, because it is intractable to construct a dataset considering diverse road conditions and defect patterns. To end this, we propose an unsupervised approach to detect road defects, using Adversarial Image-to-Frequency Transform (AIFT). AIFT adopts the unsupervised manner and adversarial learning in deriving the defect detection model, so AIFT does not require annotations for road defects. We evaluate the efficiency of AIFT using GAPs384 dataset, Cracktree200 dataset, CRACK500 dataset, and CFD dataset. The experimental results demonstrate that the proposed approach detects various road detects, and it outperforms existing state-of-the-art approaches.

5.1 Road Pavement Defect Detection

Road defect detection is one of the important studies to prevent vehicle accidents and manage the road condition effectively. All over the United States, road conditions contribute to the frequency and severity of motor vehicle accidents. Almost of third of all motor vehicle crashes are related to poor road conditions, resulting in more than two million injuries

and 22,000 fatalities [202]. Over time, as road infrastructure ages, the condition of that infrastructure steadily declines, and the volumes and severity of defects increase [203]. Therefore, there is an increasing demand for the development of road defect detection method [204], and numerous studies have been proposed in this literature.

Over the past decades, many studies have considered the use of image processing and machine learning approaches with hand-crafted features [205, 206, 207, 208, 209]. Statistical analysis [205, 207] is a classical method with long history and most popular. Acosta *et al.*,[205] and Deutschl *et al.*,[208] have proposed vision-based methods based on partial differential techniques. Chambon *et al.*,[207] presented a method based on Markovian modelling to take into account the local geometrical constraints about road cracks. Bray *et al.*,[206] utilized the classification approach using neural networks for identifying road defects. These approaches usually identify road defects using the contrast of texture information on a road surface.

However, the contrast between roads and the defects on the roads may be reduced due to the illumination conditions and the changes in weather [210, 211]. Additionally, the specification of cameras for capturing the surface of the roads also can affect the detection accuracies. Hense, it is still challenging to develop a defect detection method which can cover various road conditions in the real world using a simple image processing or machine learning methods alone [212].

Recently, various approaches [213, 214] based on deep learning have been proposed to overcome these drawbacks. Pauly *et al.*,[213] proposed a method for road defect detection employing convolutional neural networks (CNNs). Fan *et al.*,[214] proposed segmentation

method based on CNNs and apply an adaptive. These approaches need a well-annotated dataset for road defects, and also their performance may depend on scale of the given dataset. Regrettably, it is problematic in practice to construct such a dataset containing various patterns of road defects.

Development of an unsupervised method has been an important research topic in the literature. Various unsupervised approaches based on image processing and machine learning were proposed [215, 216]. However, these approaches still have an inherent weakness which is detection performance is highly dependent on camera specifications and image qualities. Recently, among the approaches based on deep learning, several studies [217, 218] have presented unsupervised methods using autoencoder [119]. These approaches take normal road images as their training samples and optimize their models in a way to minimize reconstruction errors between their input and output. These approaches recognize defects if the reconstruction errors of inputted samples are larger than a predefined threshold.

However, according to Perera *et al.*,[123] and Pidhorskyi *et al.*,[219], even though a model based on the reconstruction setting obtains a well-optimized solution, there is a possibility that the model can reconstruct samples which have not appeared in the training step. It could be a significant disadvantage in detecting road defects using the model. Due to this disadvantage, the model may produce lower error than the expectation even if it takes defect samples as their input, and it can make hard to distinguish whether this sample contains defects or not.

To tackle this issue, we present an unsupervised approach, which exploits domain transformation based on adversarial learning, to detecting road defects. The proposed approach called Adversarial Image-to-Frequency Transform (AIFT) is trained by normal road images only

and needs no annotations for defects. In contrast to other approaches [217, 218] optimizing their models by minimize reconstruction errors, AIFT is concentrated on deriving mapping function between an image-domain and a frequency-domain using adversarial manner. To demonstrate the efficiency of the proposed approach for road defect detection, we compare the proposed approach with various state-of-the-art approaches, including supervised and unsupervised methods. The experimental results show that the proposed approach outperforms existing state-of-the-art methods.

The main contributions of our work are summarized as follows:

- An unsupervised method for detecting road defects, which can provide outstanding performance without a well-annotated dataset for road defects.
- The adversarial learning for deriving the image-to-frequency mapping function. Our approach derive a nearly optimal transform model than typical approaches such as reconstruction or classification settings.
- The extensive experiments about road defect detection. The experiments include ablation analysis depending on the loss functions and comprehensive comparison with the existing state-of-the-art methods.

In the following sections, we describe the details of our approach and provide the experimental results and analysis it. We conclude this paper by summarizing our works.

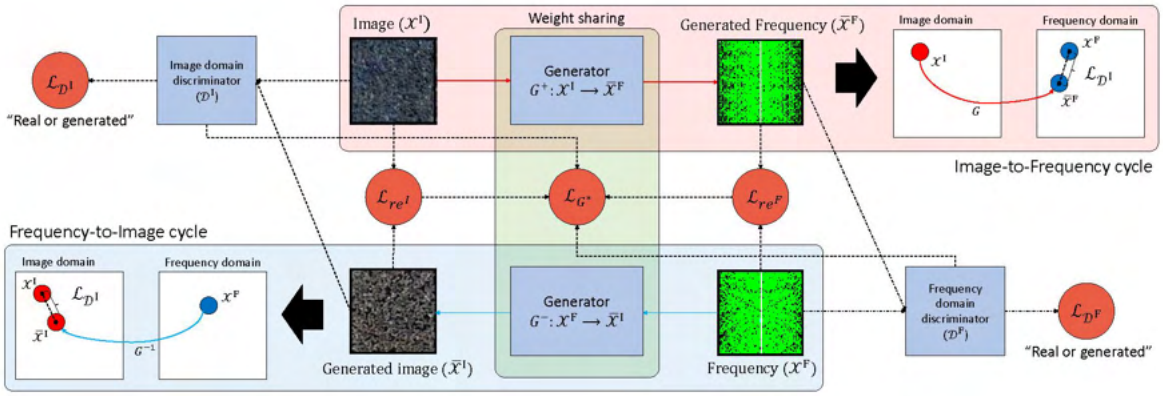


Figure 5.1: Architectural detail of the adversarial image-to-frequency transform. The blue objects denote the operation units including the generator G and the discriminators $\mathcal{D}^I$ and $\mathcal{D}^F$. The red circles indicate the loss functions corresponded to the each operation unit. The red arrow lines show the work flow for the image-to-frequency cycle $G^+ : \mathcal{X}^I \rightarrow \bar{\mathcal{X}}^F$, and the blue arrow lines represent the process of the frequency-to-image cycle $G^{-1} : \mathcal{X}^F \rightarrow \bar{\mathcal{X}}^I$. The dotted arrow lines represent the correlations of each component to the loss functions.

5.2 Adversarial Image-to-Frequency Transform

5.2.1 Image-to-frequency transformation

It is essential to derive a robust model invariant to environments in order to detect a great number of defect patterns on roads. Our method is inspired by novelty detection and abnormal event detection studies [123, 219, 220, 111], which derive a model using inlier samples only and recognize outliers by computing a likelihood or an reconstruction error. The proposed method, called Adversarial Image-to-Frequency Transform (AIFT), initially derives a transform model between image-domain and frequency-domain using normal road pavement images only. The frequency-domain corresponding to the image-domain is generated by applying Fourier transform to the given image-domain. Detecting road defects is conducted by comparing given and generated samples of each domain.

AIFT is composed of three components: Generator G, Image discriminator $\mathcal{D}^I$, Frequency discriminator $\mathcal{D}^F$, for applying adversarial learning. The original intention of adversarial

learning is to learn generative models while avoiding approximating many intractable probabilistic computations arising in other strategies *e.g.,* maximum likelihood estimation. This intention is suitable to derive an optimal model for covering the various visual patterns of road defects. The workflow of AIFT is illustrated in Fig 5.1.

The generator G plays as a role for the mapping function between image-domain $\boldsymbol{\mathcal{X}}^I = \{\mathcal{X}_i^I\}_{i=1:n}$ to frequency-domain $\boldsymbol{\mathcal{X}}^F = \{\mathcal{X}_i^F\}_{i=1:n}$ as follows, $G : \mathcal{X}^I \longleftrightarrow \mathcal{X}^F$. For the convenience of notation, we distinguish the notations of mappings for image-to-frequency $G^+ : \boldsymbol{\mathcal{X}}^I \rightarrow \boldsymbol{\mathcal{X}}^F$ and frequency-to-image $G^- : \boldsymbol{\mathcal{X}}^F \rightarrow \boldsymbol{\mathcal{X}}^I$, separately. G generate the transformed results from each domain as follows,

$$\begin{aligned} G^+(\mathcal{X}^I) &= \bar{\mathcal{X}}^F, \\ G^-(\mathcal{X}^F) &= \bar{\mathcal{X}}^I, \end{aligned} \tag{5.1}$$

where $\bar{\mathcal{X}}^F$ and $\bar{\mathcal{X}}^I$ indicate the transformed results from $\mathcal{X}^I$ and $\mathcal{X}^F$, respectively. $\bar{\mathcal{X}}^I$ and $\bar{\mathcal{X}}^F$ are conveyed to the two discriminators $\mathcal{D}^I$ and $\mathcal{D}^F$ for computing an adversarial loss. For computational-cost-effective implementation, weight sharing has employed.

The discriminators $\mathcal{D}^I$ and $\mathcal{D}^F$ are defined as follows,

$$\mathcal{D}^*(\mathcal{X}^*) = o^*, \quad o^* \in R^1, \tag{5.2}$$

where $*$ denotes the indicator to assign the discriminators $\mathcal{D}^* \in \{\mathcal{D}^I, \mathcal{D}^F\}$ depending on the types of inputs $\mathcal{X}^* \in \{\mathcal{X}^I, \mathcal{X}^F, \bar{\mathcal{X}}^I, \bar{\mathcal{X}}^F\}$. $\mathcal{D}^I$ takes $\mathcal{X}^I$ and $\bar{\mathcal{X}}^I$ as an input, and $\mathcal{D}^I$ takes $\mathcal{X}^F$ and $\bar{\mathcal{X}}^F$ as an input, respectively. o^* indicates the outputs o^I and o^F according to the types

of the inputs and the discriminators. The value of o^* can be regarded by as a likelihood to discriminate whether a given sample is truth or generated. Each component is compiled by CNNs and fully-connected neural networks and the structural details of these components are shown in Fig 5.2.

5.2.2 Adversarial learning for image-to-frequency transformation

As the workflow of AIFT shown in Fig 5.1, the generator G plays a role as a bidirectional mapping function between image-domain $\boldsymbol{\mathcal{X}}^I$ and corresponding frequency-domain $\boldsymbol{\mathcal{X}}^F$ generated from $\boldsymbol{\mathcal{X}}^I$. The underlying assumption for detecting road defects using AIFT is as follows. Since AIFT is only trained with normal road pavement images, if AIFT takes images containing defect patterns as an input, the error between the given samples and the transformed results would be larger than normal ones. Given this assumption, the prerequisite for precise road defect detection on AIFT is deriving a strict transform model between the image-domain and the frequency-domain from a given dataset for normal image samples for road pavement.

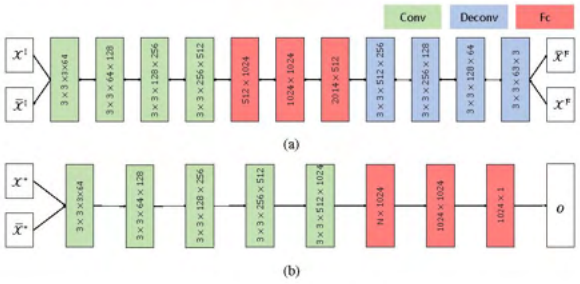


Figure 5.2: Structural details of the network models in the generator G and the discriminators $\mathcal{D}^I$ and $\mathcal{D}^F$. (a) and (b) denote the structural details of the generator G and the two discriminators $\mathcal{D}^I$ and $\mathcal{D}^F$, respectively. The green, blue, and red boxes denote the convolutional layers, the deconvolutional layers, and the fully-connected layers, respectively.

To end this, we present an adversarial transform consistency loss for training AIFT.

Adversarial transform consistency loss is defined by,

$$\begin{aligned}\mathcal{L}_{\text{ATCL}}(G, \mathcal{D}^I, \mathcal{D}^F) = & E_{\mathcal{X}^I \sim p_{\boldsymbol{\mathcal{X}^I}}}[\log \mathcal{D}^I(\mathcal{X}^I)] \\ & + E_{\mathcal{X}^F \sim p_{\boldsymbol{\mathcal{X}^F}}}[\log \mathcal{D}^F(\mathcal{X}^F)] \\ & + E_{\bar{\mathcal{X}}^F \sim p_{G^+(\boldsymbol{\mathcal{X}^I})}}[\log(1 - \mathcal{D}^F(G^+(\mathcal{X}^I)))] \\ & + E_{\bar{\mathcal{X}}^I \sim p_{G^-(\boldsymbol{\mathcal{X}^F})}}[\log(1 - \mathcal{D}^I(G^-(\mathcal{X}^F)))],\end{aligned} \tag{5.3}$$

where G tries to generate images $\bar{\mathcal{X}}^I$ and frequency samples $\bar{\mathcal{X}}^F$ via G^+ and G^- that look similar to given images $\mathcal{X}^I$ and frequencies $\mathcal{X}^F$, while $\mathcal{D}^I$ and $\mathcal{D}^F$ aim to distinguish between given samples ($\mathcal{X}^I$ and $\mathcal{X}^F$) and transformed results ($\bar{\mathcal{X}}^I$ and $\bar{\mathcal{X}}^F$).

Adversarial learning can, in theory, learn mappings G that produce outputs identically distributed as image and frequency domains, respectively [221]. However, with large enough capacity, G can map the same samples of an input domain to any random permutation of samples in the different domain, where any of the learned mappings can induce an output distribution that matches the target distribution. Thus, adversarial transform consistency loss alone may not guarantee that the learned function can map an individual input to the desired output.

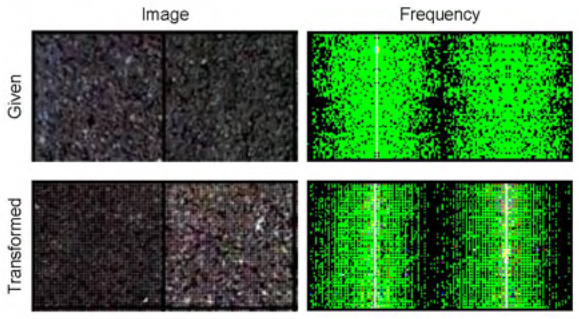


Figure 5.3: Comparison of the given and generated samples for the road pavement image and the corresponding frequency.

To further reduce the space of possible mapping functions, we utilize the reconstruction loss to optimize the generator $\mathcal{G}$. It is a common way to enforce the output of the generator to be close to the target through the minimization of the reconstruction error based on the

pixel-wise mean square error (MSE) [140, 141, 125, 112, 111]. It is calculated in the form

$$\begin{aligned}\mathcal{L}_{\text{re}}(G) = \mathbb{E}_{\mathcal{X}^I \sim p_{\boldsymbol{\mathcal{X}^I}}}[\|\mathcal{X}^F - G^+(\mathcal{X}^I)\|_2^2] \\ + E_{\mathcal{X}^F \sim p_{\boldsymbol{\mathcal{X}^F}}}[\|\mathcal{X}^I - G^-(\mathcal{X}^F)\|_2^2].\end{aligned} \tag{5.4}$$

Consequently, the total loss function is:,

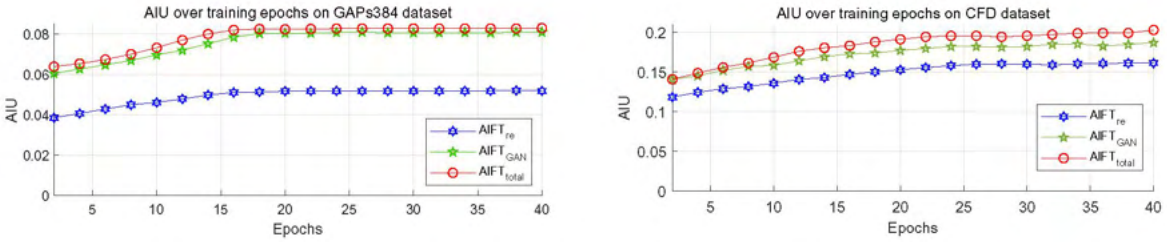
$$\mathcal{L}_{\text{total}}(G, \mathcal{D}^I, \mathcal{D}^F) = \mathcal{L}_{\text{ATCL}}(G, \mathcal{D}^I, \mathcal{D}^F) + \lambda \mathcal{L}_{\text{re}}(G) \tag{5.5}$$

where λ indicates the balancing parameter to take the weight for the reconstruction loss. Since total loss function is composed of two loss terms, we conduct ablation study to monitor the effect of each loss term in the training step.

Given the definition of above loss functions, the discriminators and the generator are trained by maximizing or minimizing corresponding loss terms expressed by,

$$\arg \min_{\theta^G} \max_{\theta^I, \theta^F} \mathcal{L}_{\text{total}}(G, \mathcal{D}^I, \mathcal{D}^F), \tag{5.6}$$

where θ^G, θ^I,and θ^F denote the parameters corresponded to the generator $\mathcal{G}$, the image discriminators $\mathcal{D}^I$, and the frequency discriminator $\mathcal{D}^F$. Fig 5.3 illustrates the examples of the given samples and the transformed results for image and frequency domains. We have conducted the ablation studies to observe the effect of each loss term in learning AIFT.



(a) (b)

Figure 5.4: The trends of AIU over the training epochs. (a) show the AIU trend over the training epochs on GAPs384 dataset, and (b) illustrate the AIU trend with respect to the training epochs on CFD dataset. The red-coloured curve ($\text{AIFT}_{\text{total}}$) denotes the AIU trend of AIFN trained by the total loss (Eq 5.5). The green-colored curve (AIFT_{GAN} a.k.a., $\text{AIFT}_{\text{ATCL}}$) indicates the AIU trend of AIFN trained by the ATCL loss (Eq 5.3) only. The blue-colored curve (AIFT_{re}) shows the AIU trend of AIF trained by the reconstruction loss (Eq 5.4).

5.2.3 Defect detection

Detecting defects on a road is straightforward. Initially, AIFT produces the frequency sample $\mathcal{X}^F$ using given an image samples $\mathcal{X}^I$. Secondly, AIFT transforms $\mathcal{X}^F$ into the image samples $\bar{\mathcal{X}}^I$ via G^-. Road defects are detected by comparing the given image sample $\mathcal{X}^I$ with the transformed result $\bar{\mathcal{X}}^I$.

Similarity metric for comparing the two samples $\mathcal{X}^I$ and $\bar{\mathcal{X}}^I$, is defined as follows,

$$d(\mathcal{X}^I, \bar{\mathcal{X}}^I) = \sum_{i,j} (\bar{x}^I_{i,j} \log \frac{\bar{x}^I_{i,j}}{m_{i,j}} - x^I_{i,j} \log \frac{x^I_{i,j}}{m_{i,j}}), \tag{5.7}$$

where $m_{i,j}$ is expectation of $x^I_{i,j}$ and $\bar{x}^I_{i,j}$. Above similarity metric is based on Jeffery divergence, which is a modified KL-divergence to take symmetric property. Euclidean distances such as $l1$-norm and $l2$-normal are not suitable as a similarity metric for images since neighboring values are not considered [145]. Jeffrey divergence is numerically stable, symmetric, and invariant to noise and input scale [146].

5.3 Experiment

5.3.1 Experimental setting

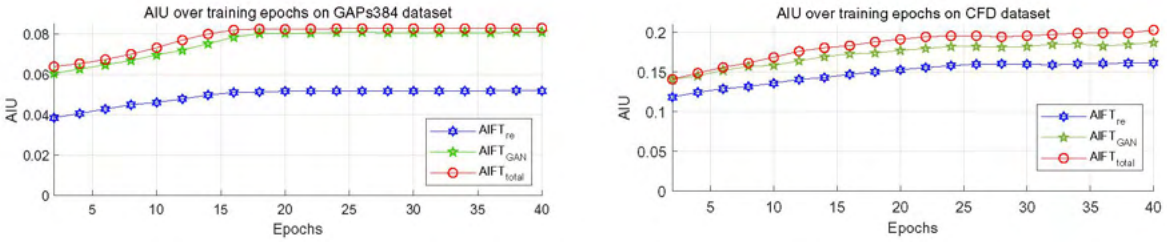
To evaluation the performance of the proposed method on road defect detection, we employ the best F-measure on the dataset for a fixed scale (ODS), the aggregate F-measure on the dataset for the best scale in each image (OIS), and AIU, which is proposed by Yang *et al.*,[3]. AIU is computed on the detection and ground truth without non-max suppression (NNS) and thinking operation, defined by, $\frac{1}{N_t}\sum_t \frac{N_{pg}^t}{N_p^t+N_g^t-N_{pg}^t}$, where N_t denotes the total number of thresholds $t \in [0.01, 0.99]$ with interval 0.01; for a given t, N_{pg}^t is the number of pixels of intersected region between the predicted and ground truth crack; N_p^t and N_g^t denote the number of pixels of predicted and ground truth crack region, respectively [3]. The Higher values of these metrics can be thought that a model provides more precise performance.

The proposed method has been evaluated on four publicly available datasets. The details of the datasets are described as follows.

GAPs384 dataset is German Asphalt Pavement Distress (GAPs) dataset presented by Eisenbach *et al.*,[1], and it is constructed to address the issue of comparability in the pavement distress domain by providing a standardized high-quality dataset of large scale. The dataset contains 1,969 gray scaled images for road defects, with various classes for defects fsuch as cracks, potholes, and inlaid patches. The resolution of images is 1,920×1,080.

Cracktree200 dataset [2] contains 206 road pavement images with 800×600 resolution, which can be categorized to various types of pavement defects. The images on this dataset are captured with some challenging issues such as shadows, occlusions, low contrast, and noise.

CRACK500 dataset is constructed by Yang *et al.*,[3]. The dataset is composed of 500



(a) (b)

Figure 5.5: The trends of AIU over the training epochs. (a) show the AIU trend over the training epochs on GAPs384 dataset, and (b) illustrate the AIU trend with respect to the training epochs on CFD dataset. The red-coloured curve ($\text{AIFT}_{\text{total}}$) denotes the AIU trend of AIFN trained by the total loss (Eq 5.5). The green-colored curve (AIFT_{GAN} a.k.a., $\text{AIFT}_{\text{ATCL}}$) indicates the AIU trend of AIFN trained by the ATCL loss (Eq 5.3) only. The blue-colored curve (AIFT_{re}) shows the AIU trend of AIF trained by the reconstruction loss (Eq 5.4).

images wity 2,000×1,500, and each image has a pixel-level annotation. The dataset is seperated by training dataset and test dataset. The training dataset consists of 1,896 images, and the test dataset is composed of 1,124 images.

CFD dataset [4] contains 118 images with 480×320 resolution. Each image has pixel-level annotation and captured by Iphone 5 with focus of 4mm aperture of $f/2.4$ and exposure time of 1/135s.

The hyperparameter setting for the best performance is as follows. The epoch size and the batch size are 50 and 64, respectively. The balancing weight for the reconstruction loss E_{re} is set by 0.1, and the critic iteration is set by 10 for the best performance. The networks are optimized by Adam *et al.*,[109]. The proposed approach has implemented with Pytorch library [1], and the experiments have conducted with GTX Titan XP and 32GB memory.

5.3.2 Ablation study

[1] Source codes are publicly available on `https://github.com/andreYoo/Adversarial_IFTN`

We have conducted an ablation study to observe the effect of the loss function terms on the performance of AIFT. We have trained AIFT using the three loss functions $\mathcal{L}_{re}$ (Eq 5.4), $\mathcal{L}_{ATCL}$ (Eq 5.3), and $\mathcal{L}_{total}$ (Eq 5.5) using GAPs384 dataset and CFD dataset, and observed AIU at every two epochs. The hyperparameter settings applied to train each model, are all same, and only the loss functions are different. Fig 5.5 shows the AIU trends of AIFTs trained by the three loss functions. Table 5.1 contains AIUs, ODSs, and OISs on GAPs384 dataset and CFD dataset. The experimental results show that AIFT trained by the total loss ($AIFT_{total}$) achieves the best performance on this experiments. As shown in Table 5.1, $AIFT_{total}$ achieves 0.083 of AIU, 0.247 of OIS, and 0.249 of ODS for GAPs384 dataset. These figures show that $AIFT_{total}$ can produce approximately 7% better performance than others. In the experiments using CFD dataset, $AIFT_{total}$ achieves 0.203 of AIU, 0.701 of OIS, and 0.732 of ODS, and these figure are all higher than that of the others.

Table 5.1: Quantitative performance comparison of the detection performance on AIFT using GAPs384 dataset and CFD dataset depending on the loss functions $\mathcal{L}_{re}$ (Eq 5.4), $\mathcal{L}_{ATCL}$ (Eq 5.3), and $\mathcal{L}_{total}$ (Eq 5.5). The bolded figures indicate the best performances on the experiments.

Model	GAPs384 dataset [1]			CFD dataset [4]		
	AIU	ODS	OIS	AIU	ODS	OIS
$AIFT_{re}$	0.052	0.181	0.201	0.152	0.562	0.572
$AIFT_{ATCL}$	0.081	0.226	0.234	0.187	0.642	0.659
$AIFT_{total}$	**0.083**	**0.247**	**0.249**	**0.203**	**0.701**	**0.732**

Notably, the overall experimental results demonstrate that the AIFTs trained by adversarial learning, can outperform the AIFT based on the reconstruction setting ($AIFT_{re}$). Not only $AIFT_{total}$, but also $AIFT_{ATCL}$ obtains the improved achievement than $AIFT_{re}$. The AIU Trends (Fig 5.5) also justify that the AIFT learnt by adversarial manners can outperform the AIFT trained by the reconstruction setting. The experimental results justify adversarial learning can improve the robustness of AIFT for detecting road defects. For efficient experiments, only

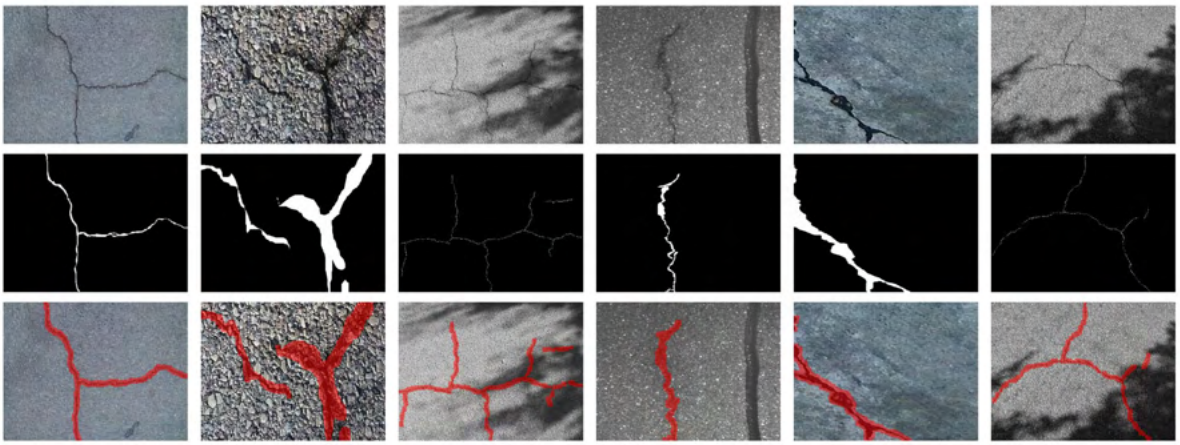


Figure 5.6: Visualization of the road defect detection results. The images on the first row represent the input images. The second row's images illustrate the ground-truths. The images on the third row denote the detection results for road defects.

Methods	S/U	GAPs384 [1]			Cracktree200 [2]			CRACK500 [3]			CFD [4]			FPS(s)
		AIU	ODS	OIS	AIU	ODS	OIS	AIU	ODS	OIS	AIU	ODS	OIS	
HED [222]	S	0.069	0.209	0.175	0.040	0.317	0.449	0.481	0.575	0.625	0.154	0.683	0.705	0.0825
RCF [223]	S	0.043	0.172	0.120	0.032	0.255	0.487	0.403	0.490	0.586	0.105	0.542	0.607	**0.079**
FCN [75]	S	0.015	0.088	0.091	0.008	0.334	0.333	0.379	0.513	0.577	0.021	0.585	0.609	0.114
CrackForest [4]	U	-	0.126	0.126	-	0.080	0.080	-	0.199	0.199	-	0.104	0.104	3.971
FPHBN [3]	S	0.081	0.220	0.231	0.041	0.517	0.579	**0.489**	**0.604**	**0.635**	0.173	0.683	0.705	0.237
AAE [224]	U	0.062	0.196	0.202	0.039	0.472	0.491	0.371	0.481	0.583	0.142	0.594	0.613	0.721
SVM [225]	S	0.051	0.132	0.162	0.017	0.382	0.391	0.362	0.418	0.426	0.082	0.3R52	0.372	0.852
ConvNet [225]	S	0.079	0.203	0.211	0.037	0.472	0.499	0.431	0.591	0.609	0.152	0.579	0.677	0.921
$AIFT_{total}$		**0.083**	**0.247**	**0.249**	**0.045**	**0.607**	**0.642**	0.478	0.549	0.561	**0.203**	**0.701**	**0.732**	1.1330

Table 5.2: Quantitative performance comparison about road defect detection using GAPs384 [1], Cracktree200 [2], CRACK500 [3], and CFD [4]. "-" means the results are not provided. The **bolded** figures indicate that the best performance among them. 'S/U' denotes whether a model focuses on *'supervised'* or *'unsupervised'* approaches. FPS indicates the execution speed of each method, and it is computed by averaging the execution speeds about all datasets.

$AIFT_{total}$ is compared with existing state-of-the-art methods.

5.3.3 Comparison with existing state-of-the-arts

We have carried out the comparison with existing state-of-the-art methods for the crack detection [222, 4, 3] and the road defect detection [225]. For the efficiency of the experiments, only $AIFT_{total}$ is compared with other methods. The performances of the existing state-of-the-art methods are referred from Yang *et al.*,[3]. Table 5.2 contains AIUs, OISs, and ODSs

on Cracktree200, GAPs384, Cracktree200, and CFD datasets. $AIFT_{total}$ has achieved state-of-the-art performance for GAPs384 dataset, Cracktree200 dataset, and CFD dataset. In the experiments using GAPs384 dataset, $AIFT_{total}$ achieves 0.083 of AIU, 0.247 of ODS, and 0.249 of OIS. These figures show that $AIFT_{total}$ outperforms than the previous state-of-the-art performance that achieved by FPHBN [3]. FPHBN obtains 0.081 of AIU, 0.220 of ODS, and 0.231 of OIS. $AIFT_{total}$ shows 3% better performances than FPHBN. The experiments on Cracktree200 dataset and CFD dataset also show that $AIFT_{total}$ surpasses other methods. $AIFT_{total}$ produces 0.045 of AIU, 0.607 of ODS, and 0.642 of OIS in the experiments using Cracktree200 dataset. Additionally, $AIFT_{total}$ achieves 0.203 of AIU, 0.701 of ODS, and 0.732 of OIS on CFD dataset. These figures are 8.8% and 2% better than the previous state-of-the-art methods approximately.

One of the interesting things is that the performance of $AIFT_{total}$ surpass the Mujeeb *et al.*,[217] and Kang *et al.*,[218], which employ methodologically similar approaches using reconstruction manner based on antoencoder. These results can be regarded as the proposed adversarial learning can provide a more effective way to learn discriminative features than the general reconstruction manner using an autoencoder. However, $AIFT_{total}$ could not obtain the highest performance on CRACK500 dataset. The state-of-the-art performance on CRACK500 dataset is achieved by FPHBN [3], and it produces 0.489 of AIU, 0.604 of ODS, and 0.635 of OIS, respectively. $AIFT_{total}$ has 0.478 of AIU, 0.549 of ODS, and 0.561 of OIS. The gaps between FPHBN and $AIFT_{total}$ are 0.011 on AIU, 0.055 on ODS, and 0.074 on OIS. However, FPHBN exploits a supervised approach, and it needs predetermined pixel-level annotations for road defects. Also, the network architecture applied to their approach is much deeper than

ours.

The overall experiments show that $\text{AIFT}_{\text{total}}$ can outperform existing state-of-the-art methods. As shown in Table 5.2, the detection performance of $\text{AIFT}_{\text{total}}$ surpasses other unsupervised methods [4, 224]. Additionally, $\text{AIFT}_{\text{total}}$ achieves outstanding detection performance in detecting defects than others based on supervised learning approaches, even $\text{AIFT}_{\text{total}}$ does not need an annotation for road defects in the training step. This may be thought that $\text{AIFT}_{\text{total}}$ is enabled to apply various practical situations in which a large-scale and well-annotated dataset can not be used. Consequently, the experimental results demonstrate that $\text{AIFT}_{\text{total}}$ can outperform existing state-of-the-art methods.

5.4 Conclusion and Discussion

We have proposed an unsupervised approach to detecting road defects, based on adversarial image-to-frequency transform. The experimental results demonstrate the proposed approach can detect various patterns of road defects without explicit annotations for road defects in the training step, and it outperforms existing state-of-the-art methods in most of the cases for experiments of road defect detection.

However, it is worth mentioning a few limitations of the proposed method. Firstly, as shown in Table 5.2, the execution speed is slower than other methods. Secondly, even though AIFT shows outstanding performance in defect detection, a large-scale dataset is essential to train AIFT. These two issues limitations are may inherent issues for existing methods based on deep learning for visual recognition studies. Our future works would be concentrated on handling these issues.

Chapter 6

Conclusion and Future works

Recently, machine learning, computer vision, and artificial intelligence have achieved considerable advancements alongside with the development of deep learning. The rapidly emerging fields of Machine Learning (ML) and Artificial Intelligence (AI) have achieved remarkable advancements in various business and industries and demand for automated understanding of massive data is higher than ever before. These advancements and needs are disrupting many traditional business and industries and promise to ultimately reorganize many aspects of daily life. Particularly, these reorganization is in progress rapidly in the industries such as the health-care, Internet of Things (IoT), autonomous driving, financial service, and urban planning, where a tremendous amount of data is being generated every second. These advancements are due to the extraordinary abilities of the weighted and cascaded non-linear kernel structure of deep learning, for extracting useful representation and modelling the complex data distribution. The structural characteristics of deep learning grants powerful generalization capacity to derived generative or discriminative models using deep learning, and it helps to overcome the limitations of past machine learning and artificial intelligence studies constrained by linearity.

6.1 Conclusion

This thesis has presented a novel generative model and learning strategies to derive more discriminative stochastic model and improve the discriminative power of anomaly detection models for identifying the outliers. Explored studies have demonstrated that any complex distribution can be understood by breaking it down into a set of small distribution and detect each small distribution as individuals' deviated from the previously learned distributions. Such approaches are validated through results of chapters 3, 4, and 5, where the novel situations are modelled in terms of outliers of the learned situations.

In particular, in section 3 we presented a GAN-based approach for abnormality detection. We proposed a generative deep learning method based on dual adversarial learning for positive and negative samples. Since our GANs are trained using only normal data, they are not able to generate abnormal data. At testing time, a local difference between the real and the generated images is used to detect possible abnormalities. We formulate the dual adversarial learning which can improve the discriminativeness of stochastic model, inspired by the triple loss on metric learnig. Differently from common generation-oriented GANs, during training we directly use the adversarial leanring not only maximizing the probabilistic relationship between output and positive sample, but also minimize the probabilistic relationship between output and negative samples. In order for this approach to be effective, we designed latent space alignment based on an adversarial learning.

Additionally, as shown in Chapter 4, this thesis also proposes an approach for automatically reinforcing the learnt representation for unseen situations in the environment where simple models are not applicable. Such methodology consists of an auxiliary information generator

and the feature reinforcing model. The model can be learned later on for further purposes such as classification, prediction or detection of abnormalities. We apply the paradigm of active learning and self-supervised learning for reinforcing the discriminativeness of learnt representation. The key contribution of the proposed method is that our method can improve the discrminativeness of learnt representation not only in the training step but also in the testing step automatically.

In Chapter 5,We proposed a GAN-based multi-domain transformation model that provide unsupervised detection method for a material defect. In our experiment, we evaluation our domain-transformation method between image and frequency. The proposed method can provide outstanding defect detection performance even a large-scale and well-categorized dataset.

6.2 Future works

My goal is to produce lifelong learning systems which reliably and efficiently adapt and expand their knowledge in response to an ever-changing world. I plan to focus on the following challenges: designing efficient methods to learn from multimodal data streams, where information comes as different modalities such as images, text, sensory data, etc.; developing online methods to evaluating the validation of data from real-time massive data stream; designing reliable and safe learning algorithms with rigorous guarantees for safety-critical systems; and providing generalization guarantee for the performance of deep neural networks trained on big datasets. I am in particular excited about applications of my techniques in domains such as medical and health-care, Internet of Things (IoT), self-driving cars,

financial services, and urban planning, where a tremendous amount of data is generated every second and demands fast and accurate analysis.

Semi-supervised Learning From Massive Multimodal Streams. In many domains, such as Internet of Things (IoT), stock exchange, computational social science and health-care, rapid streams of data are generated from various sources. For example, a typical self-driving car equipped with radar, cameras, lidar, and ultrasonic sensors, produce more than 4TB of data per day. Similarly, various sensors and smart devices in IoT applications generate a huge amount of data at a high velocity. The challenge in making use of such data is to design algorithms that can efficiently extract and fuse informative features from various sources to learn and make inference on the fly. My current research has already addressed the challenge of extracting useful information from unimodal data stream recorded from vision sensors. Moving forward, I intend to focus on efficient learning from summaries of multimodal large data streams.

Online Novelty Detection via Generative Stochastic Modelling. Training modern machine learning models on massive datasets generated by real-time contains a risk that training negative or damaged samples which can be an obstacle in optimizing the models and also it incur a substantial financial and environmental cost. One of my recent research focused on deriving a single stochastic model for dominant data from a given dataset and estimating a validation (Novelty) of data for identifying outliers, which can be considered as negative samples or abnormal samples. However, previous studies do not consider the possibility of transformation of data over time. The properties of the dominant data can be changed over time. I would like to continue this line of research to develop a new resource-efficient and fast

online approach for novelty detection.

Reliable and Safe Machine Learning via long-life Learning. The functional safety of many intelligent systems, such as autonomous robots and self-driving cars, remains largely dependent on the robustness of the underlying machine learning model. These systems are expected to operate flawlessly, and hence need to be trained on examples of all possible situational conditions. My current research demonstrated that training models on representative summaries highly improves the robustness of the learned models against noisy labels, both in theory and practice. One concrete direction I plan to pursue along these lines is to build scalable robust frameworks for safety-critical systems with very high accuracy requirements. In particular, I am interested in developing optimization methods that can provide guarantees for the quality of inference under noisy data, noisy labels, and adversarial attacks.

I strongly believe that the above research directions can advance the current state of machine learning research, and will have tremendous real-world influence. I am confident that my research and collaborations with experts in various fields, including data science, machine learning, statistics, mathematics, and theoretical computer science have equipped me with the necessary background to approach the above challenging and impactful research directions.

References

1. M. Eisenbach, R. Stricker, D. Seichter, K. Amende, K. Debes, M. Sesselmann, D. Ebersbach, U. Stoeckert, and H.-M. Gross, "How to get pavement distress detection ready for deep learning? a systematic approach," in *2017 international joint conference on neural networks (IJCNN)*, pp. 2039–2047, IEEE, 2017.

2. Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, "Cracktree: Automatic crack detection from pavement images," *Pattern Recognition Letters*, vol. 33, no. 3, pp. 227–238, 2012.

3. F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, and H. Ling, "Feature pyramid and hierarchical boosting network for pavement crack detection," *IEEE Transactions on Intelligent Transportation Systems*, 2019.

4. Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, "Automatic road crack detection using random structured forests," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3434–3445, 2016.

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

6. G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal processing magazine*, vol. 29, no. 6, pp. 82–97, 2012.

7. D. Klein and C. D. Manning, "Fast exact inference with a factored model for natural language parsing," in *Advances in neural information processing systems*, pp. 3–10, 2003.

8. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

9. C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Advances In Neural Information Processing Systems*, pp. 613–621, 2016.

10. B. Chen, W. Wang, and J. Wang, "Video imagination from a single image with transformation generation," in *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*, pp. 358–366, ACM, 2017.

11. M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

12. I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in neural information processing systems*, pp. 5767–5777, 2017.

13. Y. Cong, J. Yuan, and J. Liu, "Abnormal event detection in crowded scenes using sparse representation," *Pattern Recognition*, vol. 46, pp. 1851–1864, 2013.

14. A. Basharat, A. Gritai, and M. Shah, "Learning object motion patterns for anomaly detection and improved object detection," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8, IEEE, 2008.

15. X. Wang, X. Ma, and E. Grimson, "Unsupervised activity perception by hierarchical bayesian models," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pp. 1–8, IEEE, 2007.

16. X. Wang, K. Tieu, and E. Grimson, "Learning semantic scene models by trajectory analysis," in *European conference on computer vision*, pp. 110–123, Springer, 2006.

17. O. Javed and M. Shah, "Tracking and object classification for automated surveillance," in *European Conference on Computer Vision*, pp. 343–357, Springer, 2002.

18. A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust real-time unusual event detection using multiple fixed-location monitors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 3, pp. 555–560, 2008.

19. X. Cui, Q. Liu, M. Gao, and D. N. Metaxas, "Abnormal detection using interaction energy potentials," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011*, pp. 3161–3167, IEEE, 2011.

20. R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 935–942, 2009.

21. J. Kim and K. Grauman, "Observe locally, infer globally: a space-time mrf for detecting abnormal activities with incremental updates," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2928, 2009.

22. H. Mousavi, S. Mohammadi, A. Perina, R. Chellali, and V. Murino, "Analyzing tracklets for the detection of abnormal crowd behavior," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 148–155, 2015.

23. B. Zhao, L. Fei-Fei, and E. P. Xing, "Online detection of unusual events in videos via dynamic sparse coding," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011*, pp. 3313–3320, IEEE, 2011.

24. A. Klaser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3d-gradients,"

in *BMVC 2008-19th British Machine Vision Conference*, pp. 275–1, British Machine Vision Association, 2008.

25. L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1446–1453, IEEE Computer Society, 2009.

26. Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3449–3456, IEEE, 2011.

27. T. Xiang and S. Gong, "Incremental and adaptive abnormal behaviour detection," *Computer Vision and Image Understanding*, vol. 111, no. 1, pp. 59–73, 2008.

28. D. Du, H. Qi, Q. Huang, W. Zeng, and C. Zhang, "Abnormal event detection in crowded scenes based on structural multi-scale motion interrelated patterns," in *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, pp. 1–6, IEEE, 2013.

29. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

30. D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2147–2154, 2014.

31. P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4207–4215, 2016.

32. K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, pp. 568–576, 2014.

33. D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Computer Vision (ICCV), 2015 IEEE International Conference on*, pp. 4489–4497, IEEE, 2015.

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

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

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

37. M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," in *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pp. 733–742, IEEE, 2016.

38. H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, pp. 3551–3558, IEEE, 2013.

39. Y.-G. Jiang, Q. Dai, W. Liu, X. Xue, and C.-W. Ngo, "Human action recognition in unconstrained videos by explicit motion modeling," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3781–3795, 2015.

40. M. Jain, H. Jegou, and P. Bouthemy, "Better exploiting motion for better action recognition," in

Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pp. 2555–2562, IEEE, 2013.

41. T. Lan, Y. Wang, W. Yang, S. N. Robinovitch, and G. Mori, "Discriminative latent models for recognizing contextual group activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 34, no. 8, pp. 1549–1562, 2012.

42. Z. Fu, W. Hu, and T. Tan, "Similarity based vehicle trajectory clustering and anomaly detection," in *IEEE International Conference on Image Processing (ICIP)*, pp. 602–605, IEEE, 2005.

43. M. Nabi, *Mid-level Representation for Visual Recognition.* PhD thesis, Italian Institute of Technology, University of Genova, 2015.

44. H. Mousavi, H. K. Galoogahi, A. Perina, and V. Murino, "Detecting abnormal behavioral patterns in crowd scenarios," in *Toward Robotic Socially Believable Behaving Systems-Volume II*, pp. 185–205, 2016.

45. V. Rabaud and S. Belongie, "Counting crowded moving objects," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 705–711, IEEE Computer Society, 2006.

46. J. Rittscher, P. H. Tu, and N. Krahnstoever, "Simultaneous estimation of segmentation and shape," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 486–493, IEEE, 2005.

47. Y. Yuan, J. Fang, and Q. Wang, "Online anomaly detection in crowd scenes via structure analysis," *IEEE Transactions on Cybernetics*, vol. 45, no. 3, pp. 562–575, 2015.

48. J. S. Marques, P. M. Jorge, A. J. Abrantes, and J. M. Lemos, "Tracking groups of pedestrians in video sequences," in *Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, p. 101, 2003.

49. C. Piciarelli, C. Micheloni, and G. L. Foresti, "Trajectory-based anomalous event detection," *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, vol. 18, no. 11, pp. 1544–1554, 2008.

50. T. Zhao and R. Nevatia, "Bayesian human segmentation in crowded situations," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 459–466, IEEE Computer Society, 2003.

51. J. Shi and C. Tomasi, "Good features to track," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 593–600, IEEE, 1994.

52. B. Krausz and C. Bauckhage, "Analyzing pedestrian behavior in crowds for automatic detection of congestions," in *International Conference on Computer Vision workshops (ICCVW)*, pp. 144–149, IEEE Computer Society, 2011.

53. B. Wang, M. Ye, X. Li, F. Zhao, and J. Ding, "Abnormal crowd behavior detection using high-frequency and spatio-temporal features," *Machine Vision and Applications*, vol. 23, no. 3, pp. 501–511, 2012.

54. H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3169–3176, 2011.

55. X. Wang, X. Ma, and W. E. L. Grimson, "Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 31, no. 3, pp. 539–555, 2009.

56. S. Wu, H.-S. Wong, and Z. Yu, "A bayesian model for crowd escape behavior detection," *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, vol. 24, no. 1, pp. 85–98, 2014.

57. T. Wang and H. Snoussi, "Histograms of optical flow orientation for visual abnormal events detection," in *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 13–18, IEEE Computer Society, 2012.

58. H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 819–826, IEEE Computer Society, 2004.

59. B. Krausz and C. Bauckhage, "Loveparade 2010: Automatic video analysis of a crowd disaster," *Computer Vision and Image Understanding (CVIU)*, vol. 116, no. 3, pp. 307–319, 2012.

60. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893, IEEE, 2005.

61. L. Kratz and K. Nishino, "Tracking with local spatio-temporal motion patterns in extremely crowded scenes," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 693–700, IEEE Computer Society, 2010.

62. V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1975–1981, IEEE Computer Society, 2010.

63. D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Physical Review E, Statistical physics, plasmas, fluids, and related interdisciplinary topics*, vol. 51, no. 5, pp. 4282–4286, 1995.

64. S. Ali and M. Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–6, IEEE, 2007.

65. B. Solmaz, B. E. Moore, and M. Shah, "Identifying behaviors in crowd scenes using stability analysis for dynamical systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 34, no. 10, pp. 2064–2070, 2012.

66. S. Wu, B. E. Moore, and M. Shah, "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2054–2060, IEEE Computer Society, 2010.

67. M. Hu, S. Ali, and M. Shah, "Learning motion patterns in crowded scenes using motion flow field," 2008. International Conference on Pattern Recognition (ICPR).

68. B. Zhou, X. Wang, and X. Tang, "Random field topic model for semantic region analysis in crowded scenes from tracklets," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pp. 3441–3448, IEEE Computer Society, 2011.

69. B. Zhou, X. Wang, and X. Tang, "Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2871–2878, IEEE Computer Society, 2012.

70. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, pp. 1106–1114, 2012.

71. M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in *Proceedings of the 14th European Conference on Computer Vision (ECCV)*, pp. 525–542, 2016.

72. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla,

M. S. Bernstein, A. C. Berg, and F. Li, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

73. E. Sangineto, M. Nabi, D. Culibrk, and N. Sebe, "Self paced deep learning for weakly supervised object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 41, no. 3, pp. 712–725, 2018.

74. B. Zhou, À. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using Places Database," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems (NIPS)*, pp. 487–495, 2014.

75. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, 2015.

76. M. Ravanbakhsh, H. Mousavi, M. Nabi, M. Rastegari, and C. S. Regazzoni, "Cnn-aware binary map for general semantic segmentation," in *IEEE International Conference on Image Processing (ICIP)*, pp. 1923–1927, 2016.

77. K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems (NIPS)*, pp. 568–576, 2014.

78. C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1933–1941, 2016.

79. D. Gordon, A. Kembhavi, M. Rastegari, J. Redmon, D. Fox, and A. Farhadi, "IQA: Visual question answering in interactive environments," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4089–4098, IEEE Computer Society, 2018.

80. R. Shekhar, S. Pezzelle, A. Herbelot, M. Nabi, E. Sangineto, and R. Bernardi, "Vision and language integration: moving beyond objects," in *International Conference on Computational Semantics (IWCS)*, 2017.

81. R. Shekhar, S. Pezzelle, Y. Klimovich, A. Herbelot, M. Nabi, E. Sangineto, and R. Bernardi, "Foil it! find one mismatch between image and language caption," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 255–265, 2017.

82. A. Abad, M. Nabi, and A. Moschitti, "Self-crowdsourcing training for relation extraction," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2017.

83. A. Abad, M. Nabi, and A. Moschitti, "Autonomous crowdsourcing through human-machine collaborative learning," in *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017.

84. O. Russakovsky, L.-J. Li, and L. Fei-Fei, "Best of both worlds: human-machine collaboration for object annotation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2121–2131, 2015.

85. Z. Fang, F. Fei, Y. Fang, C. Lee, N. Xiong, L. Shu, and S. Chen, "Abnormal event detection in crowded scenes based on deep learning," *Multimedia Tools and Applications*, vol. 75, no. 22, pp. 14617–14639, 2016.

86. R. T. Ionescu, F. S. Khan, M.-I. Georgescu, and L. Shao, "Object-centric auto-encoders and dummy anomalies for abnormal event detection in video," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. CoRR arxiv available at `http://arxiv.org/abs/1812.04960`.

87. M. Ravanbakhsh, M. Nabi, H. Mousavi, E. Sangineto, and N. Sebe, "Plug-and-play cnn for

crowd motion analysis: An application in abnormal event detection," *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1689–1698, 2018.

88. M. Sabokrou, M. Fayyaz, M. Fathy, and R. Klette, "Deep-cascade: Cascading 3D deep neural networks for fast anomaly detection and localization in crowded scenes," *IEEE Transactions on Image Processing (TIP)*, vol. 26, no. 4, pp. 1992–2004, 2017.

89. M. Sabokrou, M. Fayyaz, M. Fathy, and R. Klette, "Fully convolutional neural network for fast anomaly detection in crowded scenes," *arXiv preprint CoRR*, vol. abs/1609.00866, 2016.

90. S. H. Bach, B. He, A. Ratner, and C. Ré, "Learning the structure of generative models without labeled data," in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 273–282, 2017.

91. F. Pahde, M. Puscas, J. Wolff, T. Klein, N. Sebe, and M. Nabi, "Low-shot learning from imaginary 3d model," *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 978–985, 2019.

92. S. Zhou, W. Shen, D. Zeng, M. Fang, Y. Wei, and Z. Zhang, "Spatial-temporal convolutional neural networks for anomaly detection and localization in crowded scenes," *Signal Processing: Image Communication*, vol. 47, pp. 358–368, 2016.

93. M. Sabokrou, M. Fathy, and M. Hoseini, "Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder," *Electronics Letters*, vol. 52, no. 13, pp. 1122–1124, 2016.

94. Y. Feng, Y. Yuan, and X. Lu, "Learning deep event models for crowd anomaly detection," *Neurocomputing*, vol. 219, pp. 548–556, 2017.

95. T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet, a simple deep learning baseline for image classification?," *IEEE Transaction on Image Processing (TIP)*, vol. 24, no. 12, pp. 5017–5032, 2015.

96. A. van den Oord and B. Schrauwen, "Factoring variations in natural images with deep Gaussian Mixture Models," in *Advances in Neural Information Processing Systems (NIPS) 27: Annual Conference on Neural Information Processing Systems*, pp. 3518–3526, 2014.

97. M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 733–742, IEEE Computer Society, 2016.

98. C. Lu, J. Shi, and J. Jia, "Abnormal event detection at 150 fps in matlab," in *International Conference on Computer Vision (ICCV)*, pp. 2720–2727, 2013.

99. W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6479–6488, 2018.

100. A. B. Mabrouk and E. Zagrouba, "Abnormal behavior recognition for intelligent video surveillance systems: A review," *Expert Systems with Applications*, vol. 91, pp. 480 – 491, 2018.

101. T. Hassner, Y. Itcher, and O. Kliper-Gross, "Violent flows: Real-time detection of violent crowd behavior," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1–6, IEEE Computer Society, 2012.

102. S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 1, pp. 221–231, 2013.

103. D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 922–928, IEEE, 2015.

104. Y. Zhang, L. Qin, H. Yao, and Q. Huang, "Abnormal crowd behavior detection based on social attribute-aware force model," in *IEEE International Conference on Image Processing (ICIP), 2012 19th*, pp. 2689–2692, IEEE, 2012.

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

106. G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for lvcsr using rectified linear units and dropout," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 8609–8613, IEEE, 2013.

107. J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *International Conference on Artificial Neural Networks*, pp. 52–59, Springer, 2011.

108. Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*, pp. 396–404, 1990.

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

110. M. Sabokrou, M. Fathy, M. Hoseini, and R. Klette, "Real-time anomaly detection and localiza-

tion in crowded scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 56–62, 2015.

111. J. Yu, K. C. Yow, and M. Jeon, "Joint representation learning of appearance and motion for abnormal event detection," *Machine Vision and Applications*, vol. 29, no. 7, pp. 1157–1170, 2018.

112. M. Sabokrou, M. Fayyaz, M. Fathy, Z. Moayed, and R. Klette, "Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes," *Computer Vision and Image Understanding*, vol. 172, pp. 88–97, 2018.

113. J.-X. Zhong, N. Li, W. Kong, S. Liu, T. H. Li, and G. Li, "Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1237–1246, 2019.

114. J. K. Dutta and B. Banerjee, "Online detection of abnormal events using incremental coding length," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

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

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

117. R. T. Ionescu, F. S. Khan, M.-I. Georgescu, and L. Shao, "Object-centric auto-encoders and dummy anomalies for abnormal event detection in video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7842–7851, 2019.

118. W. Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked rnn framework," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 341–349, 2017.

119. P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of machine learning research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

120. S. Lee, H. G. Kim, and Y. M. Ro, "Bman: Bidirectional multi-scale aggregation networks for abnormal event detection," *IEEE Transactions on Image Processing*, pp. 1–1, 2019.

121. W. Luo, W. Liu, D. Lian, J. Tang, L. Duan, X. Peng, and S. Gao, "Video anomaly detection with sparse coding inspired deep neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019.

122. Y. Liu, C.-L. Li, and B. Póczos, "Classifier two sample test for video anomaly detections.," in *BMVC*, p. 71, 2018.

123. P. Perera, R. Nallapati, and B. Xiang, "Ocgan: One-class novelty detection using gans with constrained latent representations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2898–2906, 2019.

124. M. Ravanbakhsh, M. Nabi, E. Sangineto, L. Marcenaro, C. S. Regazzoni, and N. Sebe, "Abnormal event detection in videos using generative adversarial nets," in *IEEE International Conference on Image Processing (ICIP)*, pp. 1577–1581, 2017.

125. W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection–a new baseline," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6536–6545, 2018.

126. T.-N. Nguyen and J. Meunier, "Anomaly detection in video sequence with appearance-motion correspondence," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

127. R. Nawaratne, D. Alahakoon, D. De Silva, and X. Yu, "Spatiotemporal anomaly detection using deep learning for real-time video surveillance," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2019.

128. K. Xu, T. Sun, and X. Jiang, "Video anomaly detection and localization based on an adaptive intra-frame classification network," *IEEE Transactions on Multimedia*, pp. 1–1, 2019.

129. L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.

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

131. M. Ravanbakhsh, M. Nabi, H. Mousavi, E. Sangineto, and N. Sebe, "Plug-and-play cnn for crowd motion analysis: An application in abnormal event detection," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1689–1698, IEEE, 2018.

132. M. Ravanbakhsh, E. Sangineto, M. Nabi, and N. Sebe, "Training adversarial discriminators for cross-channel abnormal event detection in crowds," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1896–1904, IEEE, 2019.

133. P. Molchanov, S. Gupta, K. Kim, and J. Kautz, "Hand gesture recognition with 3d convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 1–7, 2015.

134. A. Diba, A. M. Pazandeh, and L. Van Gool, "Efficient two-stream motion and appearance 3d cnns for video classification," *arXiv preprint arXiv:1608.08851*, 2016.

135. F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

136. D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based cnn with improved triplet loss function," in *Proceedings of the iEEE conference on computer vision and pattern recognition*, pp. 1335–1344, 2016.

137. S. Sankaranarayanan, A. Alavi, and R. Chellappa, "Triplet similarity embedding for face verification," *arXiv preprint arXiv:1602.03418*, 2016.

138. J. Zhu, H. Zeng, J. Huang, S. Liao, Z. Lei, C. Cai, and L. Zheng, "Vehicle re-identification using quadruple directional deep learning features," *IEEE Transactions on Intelligent Transportation Systems*, 2019.

139. W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: a deep quadruplet network for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 403–412, 2017.

140. X. Ying, H. Guo, K. Ma, J. Wu, Z. Weng, and Y. Zheng, "X2ct-gan: Reconstructing ct from biplanar x-rays with generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10619–10628, 2019.

141. Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "Finding tiny faces in the wild with generative adversarial network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 21–30, 2018.

142. S. Yan, J. S. Smith, W. Lu, and B. Zhang, "Abnormal event detection from videos using a two-stream recurrent variational autoencoder," *IEEE Transactions on Cognitive and Developmental Systems*, pp. 1–1, 2018.

143. S. Yan, Z. Li, Y. Xiong, H. Yan, and D. Lin, "Convolutional sequence generation for skeleton-based action synthesis," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4394–4402, 2019.

144. S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "Mocogan: Decomposing motion and content for video generation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1526–1535, 2018.

145. Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International journal of computer vision*, vol. 40, no. 2, pp. 99–121, 2000.

146. J. Puzicha, T. Hofmann, and J. M. Buhmann, "Non-parametric similarity measures for unsupervised texture segmentation and image retrieval," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 267–272, IEEE, 1997.

147. H. T. Tran and D. Hogg, "Anomaly detection using a convolutional winner-take-all autoencoder," in *Proceedings of the British Machine Vision Conference 2017*, British Machine Vision Association, 2017.

148. R. Tudor Ionescu, S. Smeureanu, B. Alexe, and M. Popescu, "Unmasking the abnormal events in video," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2895–2903, 2017.

149. L. Wang, F. Zhou, Z. Li, W. Zuo, and H. Tan, "Abnormal event detection in videos using

hybrid spatio-temporal autoencoder," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, pp. 2276–2280, IEEE, 2018.

150. H. Vu, T. D. Nguyen, T. Le, W. Luo, and D. Phung, "Robust anomaly detection in videos using multilevel representations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 5216–5223, 2019.

151. P. Wu, J. Liu, and F. Shen, "A deep one-class neural network for anomalous event detection in complex scenes," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2019.

152. E. Epaillard and N. Bouguila, "Variational bayesian learning of generalized dirichlet-based hidden markov models applied to unusual events detection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, pp. 1034–1047, April 2019.

153. W. Chu, H. Xue, C. Yao, and D. Cai, "Sparse coding guided spatiotemporal feature learning for abnormal event detection in large videos," *IEEE Transactions on Multimedia*, vol. 21, pp. 246–255, Jan 2019.

154. S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

155. J. Wang and A. Cherian, "Gods: Generalized one-class discriminative subspaces for anomaly detection," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

156. P. Schroeder, M. Meyers, S. Kostyniuk, Lidia, *et al.*, "National survey on distracted driving attitudes and behaviors-2012," Tech. Rep. DOT HS 811 729, United States. National Highway Traffic Safety Administration, Washington, DC, 2013.

157. G. Maycock, "Sleepiness and driving: the experience of uk car drivers," *Journal of Sleep Research*, vol. 5, no. 4, pp. 229–231, 1996.

158. F. Sagberg, "Road accidents caused by drivers falling asleep," *Accident Analysis & Prevention*, vol. 31, no. 6, pp. 639–649, 1999.

159. A. I. Pack, A. M. Pack, E. Rodgman, A. Cucchiara, D. F. Dinges, and C. W. Schwab, "Characteristics of crashes attributed to the driver having fallen asleep," *Accident Analysis & Prevention*, vol. 27, no. 6, pp. 769–775, 1995.

160. I. Garcia, S. Bronte, L. M. Bergasa, J. Almazán, and J. Yebes, "Vision-based drowsiness detector for real driving conditions," in *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pp. 618–623, IEEE, 2012.

161. R. O. Mbouna, S. G. Kong, and M.-G. Chun, "Visual analysis of eye state and head pose for driver alertness monitoring," *IEEE transactions on intelligent transportation systems*, vol. 14, no. 3, pp. 1462–1469, 2013.

162. P. Wang and L. Shen, "A method of detecting driver drowsiness state based on multi-features of face," in *Image and Signal Processing (CISP), 2012 5th International Congress on*, pp. 1171–1175, IEEE, 2012.

163. K. Minkov, S. Zafeiriou, and M. Pantic, "A comparison of different features for automatic eye blinking detection with an application to analysis of deceptive behavior," in *Communications Control and Signal Processing (ISCCSP), 2012 5th International Symposium on*, pp. 1–4, IEEE, 2012.

164. A. Panning, A. Al-Hamadi, and B. Michaelis, "A color based approach for eye blink detection

in image sequences," in *Signal and Image Processing Applications (ICSIPA), 2011 IEEE International Conference on*, pp. 40–45, IEEE, 2011.

165. Y. Kurylyak, F. Lamonaca, and G. Mirabelli, "Detection of the eye blinks for human's fatigue monitoring," in *Medical Measurements and Applications Proceedings (MeMeA), 2012 IEEE International Symposium on*, pp. 1–4, IEEE, 2012.

166. M. Suzuki, N. Yamamoto, O. Yamamoto, T. Nakano, and S. Yamamoto, "Measurement of driver's consciousness by image processing-a method for presuming driver's drowsiness by eye-blinks coping with individual differences," in *2006 IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, pp. 2891–2896, IEEE, 2006.

167. R. N. Khushaba, S. Kodagoda, S. Lal, and G. Dissanayake, "Driver drowsiness classification using fuzzy wavelet-packet-based feature-extraction algorithm," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 121–131, 2011.

168. M. Patel, S. Lal, D. Kavanagh, and P. Rossiter, "Applying neural network analysis on heart rate variability data to assess driver fatigue," *Expert systems with Applications*, vol. 38, no. 6, pp. 7235–7242, 2011.

169. Y. Tran, A. Craig, N. Wijesuriya, and H. Nguyen, "Improving classification rates for use in fatigue countermeasure devices using brain activity," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 4460–4463, IEEE, 2010.

170. C. Papadelis, Z. Chen, C. Kourtidou-Papadeli, P. D. Bamidis, I. Chouvarda, E. Bekiaris, and N. Maglaveras, "Monitoring sleepiness with on-board electrophysiological recordings for preventing sleep-deprived traffic accidents," *Clinical Neurophysiology*, vol. 118, no. 9, pp. 1906–1922, 2007.

171. T. Ersal, H. J. Fuller, O. Tsimhoni, J. L. Stein, and H. K. Fathy, "Model-based analysis and classification of driver distraction under secondary tasks," *IEEE transactions on intelligent transportation systems*, vol. 11, no. 3, pp. 692–701, 2010.

172. J. H. Yang, Z.-H. Mao, L. Tijerina, T. Pilutti, J. F. Coughlin, and E. Feron, "Detection of driver fatigue caused by sleep deprivation," *IEEE Transactions on systems, man, and cybernetics-part A: Systems and humans*, vol. 39, no. 4, pp. 694–705, 2009.

173. C. C. Liu, S. G. Hosking, and M. G. Lenné, "Predicting driver drowsiness using vehicle measures: Recent insights and future challenges," *Journal of safety research*, vol. 40, no. 4, pp. 239–245, 2009.

174. Y. Takei and Y. Furukawa, "Estimate of driver's fatigue through steering motion," in *2005 IEEE international conference on systems, man and cybernetics*, vol. 2, pp. 1765–1770, Ieee, 2005.

175. T. Wakita, K. Ozawa, C. Miyajima, K. Igarashi, I. Katunobu, K. Takeda, and F. Itakura, "Driver identification using driving behavior signals," *IEICE TRANSACTIONS on Information and Systems*, vol. 89, no. 3, pp. 1188–1194, 2006.

176. D. F. Dinges and R. Grace, "Perclos: A valid psychophysiological measure of alertness as assessed by psychomotor vigilance," *US Department of Transportation, Federal Highway Administration*, no. FHWA-MCRT-98-006, 1998.

177. K. Dwivedi, K. Biswaranjan, and A. Sethi, "Drowsy driver detection using representation learning," in *Advance Computing Conference (IACC), 2014 IEEE International*, pp. 995–999, IEEE, 2014.

178. R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in

Image Processing. 2002. Proceedings. 2002 International Conference on, vol. 1, pp. I–I, IEEE, 2002.

179. A. D. McDonald, J. D. Lee, C. Schwarz, and T. L. Brown, "A contextual and temporal algorithm for driver drowsiness detection," *Accident Analysis & Prevention*, vol. 113, pp. 25–37, 2018.

180. R. Sayed and A. Eskandarian, "Unobtrusive drowsiness detection by neural network learning of driver steering," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 215, no. 9, pp. 969–975, 2001.

181. J. Krajewski, M. Golz, and D. Sommer, "Detecting sleepy drivers by pattern recognition based analysis of steering wheel behaviour," *Der Mensch im Mittelpunkt technischer Systeme*, pp. 288–291, 2009.

182. M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

183. R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 142–158, 2016.

184. S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

185. X. Qi, C.-G. Li, G. Zhao, X. Hong, and M. Pietikäinen, "Dynamic texture and scene classification by transferring deep image features," *Neurocomputing*, vol. 171, pp. 1230–1241, 2016.

186. Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based

action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1110–1118, 2015.

187. J. Yu, S. Park, S. Lee, and M. Jeon, "Representation learning, scene understanding, and feature fusion for drowsiness detection," in *Computer Vision – ACCV 2016 Workshops* (C.-S. Chen, J. Lu, and K.-K. Ma, eds.), (Cham), pp. 165–177, Springer International Publishing, 2017.

188. S. Hong, J. Oh, H. Lee, and B. Han, "Learning transferrable knowledge for semantic segmentation with deep convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3204–3212, 2016.

189. Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Learning deep representation for face alignment with auxiliary attributes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 5, pp. 918–930, 2016.

190. B. B. Le Cun, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*, Citeseer, 1990.

191. R. Memisevic, "Learning to relate images," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1829–1846, 2013.

192. S. Hong, J. Oh, B. Han, and H. Lee, "Learning transferrable knowledge for semantic segmentation with deep convolutional neural network," *arXiv preprint arXiv:1512.07928*, 2015.

193. K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," *arXiv preprint arXiv:1502.03044*, vol. 2, no. 3, p. 5, 2015.

194. S. Abtahi, M. Omidyeganeh, S. Shirmohammadi, and B. Hariri, "Yawdd: A yawning detection dataset," in *Proceedings of the 5th ACM Multimedia Systems Conference*, pp. 24–28, ACM, 2014.

195. S. S. Farfade, M. J. Saberian, and L.-J. Li, "Multi-view face detection using deep convolutional neural networks," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pp. 643–650, ACM, 2015.

196. Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

197. O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition.," in *BMVC*, vol. 1, p. 6, 2015.

198. J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2625–2634, 2015.

199. S. Park, F. Pan, S. Kang, and C. D. Yoo, "Driver drowsiness detection system based on feature representation learning using various deep networks,"

200. K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pp. 5353–5360, IEEE, 2015.

201. A. Notchenko, E. Kapushev, and E. Burnaev, "Sparse 3d convolutional neural networks for large-scale shape retrieval," *arXiv preprint arXiv:1611.09159*, 2016.

202. E. Zaloshnja and T. R. Miller, "Cost of crashes related to road conditions, united states, 2006,"

in *Annals of Advances in Automotive Medicine/Annual Scientific Conference*, vol. 53, p. 141, Association for the Advancement of Automotive Medicine, 2009.

203. T. A. Carr, M. D. Jenkins, M. I. Iglesias, T. Buggy, and G. Morison, "Road crack detection using a single stage detector based deep neural network," in *2018 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS)*, pp. 1–5, IEEE, 2018.

204. Z. Hadavandsiri, D. D. Lichti, A. Jahraus, and D. Jarron, "Concrete preliminary damage inspection by classification of terrestrial laser scanner point clouds through systematic threshold definition," *ISPRS International Journal of Geo-Information*, vol. 8, no. 12, p. 585, 2019.

205. J. A. Acosta, J. L. Figueroa, and R. L. Mullen, "Low-cost video image processing system for evaluating pavement surface distress," *Transportation research record*, no. 1348, 1992.

206. J. Bray, B. Verma, X. Li, and W. He, "A neural network based technique for automatic classification of road cracks," in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pp. 907–912, IEEE, 2006.

207. S. Chambon, C. Gourraud, J. M. Moliard, and P. Nicolle, "Road crack extraction with adapted filtering and markov model-based segmentation: introduction and validation," 2010.

208. E. Deutschl, C. Gasser, A. Niel, and J. Werschonig, "Defect detection on rail surfaces by a vision based system," in *IEEE Intelligent Vehicles Symposium, 2004*, pp. 507–511, IEEE, 2004.

209. C. Koch and I. Brilakis, "Pothole detection in asphalt pavement images," *Advanced Engineering Informatics*, vol. 25, no. 3, pp. 507–515, 2011.

210. Y. Sun, E. Salari, and E. Chou, "Automated pavement distress detection using advanced image processing techniques," in *2009 IEEE International Conference on Electro/Information Technology*, pp. 373–377, IEEE, 2009.

211. C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Advanced Engineering Informatics*, vol. 29, no. 2, pp. 196–210, 2015.

212. M. Baygin and M. Karakose, "A new image stitching approach for resolution enhancement in camera arrays," in *2015 9th International Conference on Electrical and Electronics Engineering (ELECO)*, pp. 1186–1190, IEEE, 2015.

213. L. Pauly, D. Hogg, R. Fuentes, and H. Peel, "Deeper networks for pavement crack detection," in *Proceedings of the 34th ISARC*, pp. 479–485, IAARC, 2017.

214. R. Fan, M. J. Bocus, Y. Zhu, J. Jiao, L. Wang, F. Ma, S. Cheng, and M. Liu, "Road crack detection using deep convolutional neural network and adaptive thresholding," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, pp. 474–479, IEEE, 2019.

215. I. Abdel-Qader, S. Pashaie-Rad, O. Abudayyeh, and S. Yehia, "Pca-based algorithm for unsupervised bridge crack detection," *Advances in Engineering Software*, vol. 37, no. 12, pp. 771–778, 2006.

216. H. Oliveira and P. L. Correia, "Automatic road crack detection and characterization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 155–168, 2012.

217. A. Mujeeb, W. Dai, M. Erdt, and A. Sourin, "One class based feature learning approach for defect detection using deep autoencoders," *Advanced Engineering Informatics*, vol. 42, p. 100933, 2019.

218. G. Kang, S. Gao, L. Yu, and D. Zhang, "Deep architecture for high-speed railway insulator surface defect detection: Denoising autoencoder with multitask learning," *IEEE Transactions on Instrumentation and Measurement*, 2018.

219. S. Pidhorskyi, R. Almohsen, and G. Doretto, "Generative probabilistic novelty detection with adversarial autoencoders," in *Advances in Neural Information Processing Systems*, pp. 6822–6833, 2018.

220. J. Yu, S. Park, S. Lee, and M. Jeon, "Driver drowsiness detection using condition-adaptive representation learning framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 11, pp. 4206–4218, 2018.

221. J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

222. S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 1395–1403, 2015.

223. Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai, "Richer convolutional features for edge detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3000–3009, 2017.

224. A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv preprint arXiv:1511.05644*, 2015.

225. L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *2016 IEEE international conference on image processing (ICIP)*, pp. 3708–3712, IEEE, 2016.

Acknowledgements

I'd like to appreciate many people who helped me along my path to writing this thesis. First of all, I would like to thank my parents, Yong-sung Yu and Ae-Young Park, for raising me to value education. it is because of their never-ending support that I have had the chance to progress in life. Their dedication to my education provided the foundation for my studies.

I'd especially like to thank my thesis advisor, Prof. Moongu Jeon, for taking me under his wing, and for running a lab where so many researchers are so free to explore creative ideas. I'd also like to thank my lab members, specially Yongsang Yoon and Younkwan Lee, for all of the respects and knowledge they have shared with me.

I'd like to thank several people at Curtin University, who helped me in getting me adapted in exchange student life in Australia, including Prof. Ba-Tuong Vo and Prof Ba-Ngu Vo. Without them, it's hard to imagine how I would have survived the foreign student experience.

In addition to professors and coworkers mentioned above, I'd especially like to thank my old friends, Hyeontaek Oh, Ung Park, Jeongyoon Kim and Myeong Nam Kim, were a pleasure to serve my life and give me a great number of creative inspirations with. Several people were very supportive in my personal life during the past four years. Finally, I'd like to thank my girlfriend Jinghuanyan Liu for all of her support and understanding, and the many advise she's made to let me continue pursuing my research.