Deep Learning Anomaly Detection for Drone-based Surveillance

Slim Hamdi

Thèse
de doctorat
de l'UTT

Slim HAMDI

Deep Learning Anomaly Detection for Drone-based Surveillance

Champ disciplinaire :
Sciences pour l'Ingénieur

2021TROY0026 **Année 2021**

Thèse en cotutelle avec l'Université de Sfax - Tunisie

utt
UNIVERSITÉ DE TECHNOLOGIE
TROYES

THESE

pour l'obtention du grade de

DOCTEUR
de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

en SCIENCES POUR L'INGENIEUR

Spécialité : OPTIMISATION ET SURETE DES SYSTEMES

présentée et soutenue par

Slim HAMDI

le 7 septembre 2021

Deep Learning Anomaly Detection for Drone-based Surveillance

JURY

M. François SEPTIER	PROFESSEUR DES UNIVERSITES	Président
Mme Najoua BEN AMARA	PROFESSEURE DES UNIVERSITES	Rapporteure
M. Hassen DRIRA	MAITRE DE CONFERENCES - HDR	Rapporteur
Mme Yousra BEN JEMAA	PROFESSEURE DES UNIVERSITES	Examinatrice
M. Mohamed ABID	PROFESSEUR DES UNIVERSITES	Directeur de thèse
M. Hichem SNOUSSI	PROFESSEUR DES UNIVERSITES	Directeur de thèse

Personnalité invitée

M. Kais LOUKIL	MAITRE DE CONFERENCES (Tunisie)

UNIVERSITY OF TECHNOLOGY OF TROYES

NATIONAL ENGINEERING SCHOOL OF SFAX

Abstract

Deep Learning anomaly detection for drone based surveillance

Civil security is the set of methods implemented by a State or an organization to protect civilian populations, as well as their property and activities, in times of war, crisis, and peace, against risks or threats of any kind. Moreover, it consists of ensuring the safety of people against all types of natural risks such as fires or against various threats that could endanger their lives, as well as that of their property or activities (acts of terrorism, acts of vandalism, etc.). In recent years, the use of drones for surveillance tasks has been on the rise worldwide. So, The number of cameras that must be analyzed increases and the efficiency and accuracy of human operators have reached their limits. Moreover, in the context of anomaly detection, only normal events are available for the learning process. Therefore, the implementation of a deep learning method in unsupervised mode to solve this problem becomes fundamental. In this thesis, we have proposed many deep learning architectures capable of detecting abnormal events with high performance.

Acknowledgements

This thesis would not have been possible without the contributions of many people. I would like to sincerely thank them for their help, support, advice and time. I hope that all of them can find themselves in these few lines.

I would like to thank Mrs. Essoukri Ben Amara Najoua, Professor at National Engineering School of Sousse (ENISo), University of Sousse, in Tunisia and Mr. Hassen Drira is an Associate Professor (HDR) of Computer Science at Institut Mines-Télécom (IMT) Lille Douai, in France for agreeing to report this work. I would like to thank also M. François SEPTIER Full Professor at Universite Bretagne Sud and Mme. Yousra Ben Jemaa Full Professor at National Engineering School of Sfax (ENIS) for having accepted to examine my work.

I would also like to address my sincere thanks to Mr. Hichem SNOUSSI, professor at the University of Technology of Troyes, Mr. Mohamed ABID and Mr. Kais LOUKIL for having accompanied, advised and supported me during these three years of thesis. In addition to the exceptional quality of your supervisions, your kindnesses and your goodnesses made of you more than supervisors, but mentors in my eyes.

A big thank you also to all my colleagues Soufien, Zied, Samir, Nacef, Ronghua, Charbel, Marie, Amine, Marwa and Mondher for their advice and their precious help. I would like to tell you that I had a lot of pleasure to work with you, I will miss our animated discussions around the coffee machine.

I also thank all the people of the UTT, I think particularly of Pascale, Isabelle, Véronique and Bernadette for their patience and their help for the many administrative steps which accompanied these three years of thesis. I also thank Jean Philippe of the CRI for his help and his reactivity to any test.

Thank you Ilhem for all that you have given me, there are no words strong enough to express my gratitude to you, your help and support have been invaluable.

Of course, I would like to thank my parents, my sisters and brothers-in-law, and all my family members. Thanks to your devotion, I have been able to develop myself, complete my studies and complete this thesis.

Contents

List of Figures

List of Tables

For/Dedicated to/To my...

Chapter 1

Introduction

Civil security is the set of methods and technologies implemented by a State or an organization to protect civilian populations, as well as their property and activities, in times of war, crisis and peace, against risks or any kind of threats. Moreover, it consists of ensuring the security of individuals against all types of risks that may endanger their life, such as their property or their activities (acts of terrorism, acts of vandalism, etc.). In order to prevent and curb "criminal" behavior and acts in the public space, thanks to its low cost of equipment and reliability, drone video surveillance has for a few years become one of the available ways to ensure the safety and security of people and their goods. It is a system of closed-circuit television that describes a whole variety of video surveillance technologies, this system is connecting one or more drone video cameras in closed circuit; the captured images are sent to a central television screen or can be automatically processed and/or viewed and then archived or destroyed. These types of systems are widely used in many applications such law enforcement, building security, and route analysis. On the other hand, the necessity for an effective control of private or public places such as airports, train stations, shopping malls, crowded sports halls, military facilities is increasing, Popoola and Wang, 2012. Especially, Unmanned Aerial Vehicles (UAVs) have proved to be efficient for surveillance missions in different nature such as tracking and detection; area monitoring, automatic fire measurement, agricultural surveillance, and even in industry, etc. This important value comes not only from the capacity of an UAV to monitor hazardous areas, but also from its cost-effectiveness compared to an entire installation of fixed cameras (Figure 1.1). In the past, traditional surveillance systems relied on network cameras monitored by a human operator who must be aware of the actions of the people in the monitored scene. With the increase in the number of cameras to be analyzed, the efficiency and accuracy of human operators have reached its limits, Keval and Sasse, 2006. For example, in, Dee, 2008, the author proves that an operator can miss 60% of target events when he is viewing 9 or more displays. Then, the processing of this large amount of data with traditional systems is very difficult to handle and sometimes almost impossible to achieve its objectives. In addition, it is expensive and requires a significant number of human beings to perform surveillance.

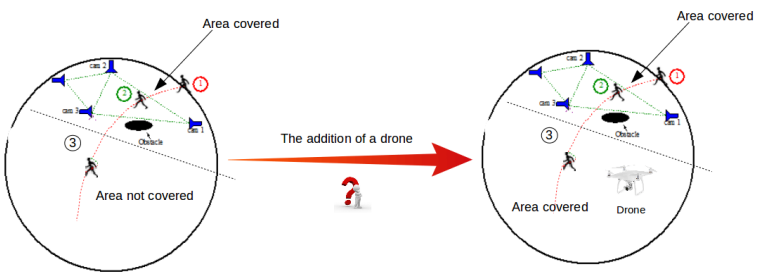


FIGURE 1.1: Using drones for surveillance

Many factors such as the fatigue and weariness caused by watching many surveillance videos for long time, the monotone of surveillance videos containing mostly normal and repetitive events and the human attention limitation, and the unexpected aspect of abnormal events, can significantly reduce the effectiveness of video surveillance systems and may lead to significant security violations. A possible solution to this problem would be the use of intelligent video surveillance systems. These systems must be capable of analyzing and modeling the normal behavior of a monitored scene and detect any abnormal behavior that could represent a security risk. In recent years, considerable technological advances in the fields of machine learning and computer vision have made possible for machines to perform certain tasks automatically. Some of them are classical machine learning methods: image classification, He et al., 2015, facial recognition, Taigman et al., 2014, human pose estimation, Toshev and Szegedy, 2013, Natural language processing, Conneau et al., 2016, automatic speech recognition, Amodei et al., 2015, and even more atypical tasks ; machine translation systems, Wu et al., 2016, lip reading, Chung et al., 2016, Negotiation Agents for E-Commerce, Barenji et al., 2019 and automatic generation of software code, Beltramelli, 2017. Deep Learning (DL) is a sub-domain of Machine Learning (ML), it aims at learning high-level abstractions in data using multi-level architectures. These different levels are obtained by stacking multiple non-linear transformation modules. Each module transforms the data at a different level until an adapted representation is obtained that allows the target task to be performed. Deep learning has overtook the traditional models in some cases of application and it has made possible to design effective pattern recognition systems without in-depth expertise on the targeted elements. Actually, the most effective deep learning methods are based on supervised learning, using large labeled databases containing samples from different classes. To benefit from these learning materials in an intelligent monitoring system, a large amount of training data representative of both normal and abnormal events must be available. However, there are many obstacles to the creation of such databases, for example we can cite the following:

- The contextual aspect of the event. Indeed the aspect of an event is closely linked to its context, an abnormal event in one scene can be normal in another. This point makes it almost impossible to design common databases that can be used in an uniform manner for different scenes.
- Risks and variability to reproduce some abnormal events make it impossible to identify and generate enough training samples.

1.1 Contributions

It is in this particularly complex context that this thesis is positioned. We will explore various strategies to develop solutions that will allow us to both exploit the potential of deep learning and avoid the creation of large labelled databases. We aim to develop methods for detection and localization of abnormal drone-video-based events using deep learning within the constraints related to the unavailability of training samples representative of abnormal events. The contributions of this thesis are in accordance with the major objective which is the adjustment and development of new architectures based on deep learning for the detection of abnormal events by a drone camera. These contributions are articulated around the following four points:

- First, we propose an efficient method based on deep learning and handcrafted spatio-temporal feature extraction for anomaly detection using a pre-trained CNN (convolution neural network) and HOF (Histogram of Optical Flow) features. Abnormal motion

is picked by relative thresholding. One-class SVM is trained with spatial features for robust classification of abnormal shapes. Then, a fusion module allows to get the final detection decision. Moreover, a decision function is applied to correct the false alarms and the miss detections. This method has a good performance in term of accuracy but it still insufficient for abnormal motion detection due to the descriptiveness of HOF features. Then, for all following methods we propose to use only deep learning architectures.

- Second, we propose an unsupervised method based on a new architecture of deep convolutional auto-encoders (CAEs) aimed at extracting a compact Spatio- temporal feature to be used for anomaly detection. Our deep CAEs are constructed by adding deconvolutions layers to the CNN VGG 16 network. The first CAE is trained on the original frames to extract a good descriptor of shapes and the second CAE is learned using optical flow images in order to encode motion between frames. For this purpose, we define two loss functions, compactness loss and representativeness loss for training our CAEs architectures. The purpose is not only to enhance the representativeness of learned features but also to ensure their tightness when encoding normal images. The shape and motion descriptors are then combined by applying a PCA (Principal Component Analyser) which further reduce the dimensionality. A Gaussian classifier is finally applied for abnormal event detection. Our method has improved the performance in terms of reliability and accuracy of abnormal event detection compared the first method. However this architecture does not capable to localizing the anomaly and the performance in term of abnormal motion detection still not very efficient due to using 2D convolution neural network.

- Third, we propose to use a new architecture based on 3D Fully Convolutional Networks (3D-FCN) to extract robust representations able to describe the shapes and movements that can occur in a monitored scene. The learned 3D-FCNs are obtained by training two Convolutional Auto-Encoders (CAEs) and extracting the encoder part of each of them. The first CAE is trained with sequences of consecutive frames to extract spatio-temporal features. The second is learned to reconstruct optical flow images from the original images, which provides a better description of the movement. We enhance our architecture with a Gaussian classifier in order to detect abnormal spatio-temporal events that could present a security risk.

  However, the optical flow computation is time consuming and prevent a fast abnormal event detection. Moreover, this method is not end-to-end architecture and need to calculate Gaussian classifier in order to detect anomaly.

- Finally, we propose an end-to-end a new architecture capable to generate optical flow images from original images and extract compact spatio-temporal features for anomaly detection purpose. It is trained with a custom loss function as a sum of three terms, Reconstruction loss (R_l), Generation loss (G_l) and Compactness loss (C_l) to ensure an effective deep-one class classification. Moreover, we propose to minimize the effect of drone motion in video processing by applying background subtraction on optical flow images. We have tested our method on very complex datasets Mini-drone video Bonetto et al., 2015 and we have achieved a good results in comparison to the existing technique with an AUC = 85.3.

1.2 Layout of the thesis

The thesis is organized as follows.

Dans le chapitre 2, L'état de l'art des méthodes de détection des anomalies et de reconnaissance des événements est présenté. Deux composantes principales, l'abstraction et la modélisation des événements, sont identifiées..

Le chapitre 3 regroups the two first contributions. Then , it is dedicated to transfer learning in the context of Convolutional Neural Networks. We start by introducing different learning methods used such as the CNN Convolutional Neural Network, the FCN Fully-Connected Network , the SVM Support Vector Machine and HOF (Histogram of Optical Flow), before presenting the first method and detailing its different components. We complete the list of previously introduced Convolution Auto-encoder (CAE) learning materials as well as the Mahalanobis distance before introducing our second method. The second architecture come to enhance the first method.

Our Chapter4 contains the third and the fourth contributions. So, it is dedicated to custom unsupervised neural networks applied to the detection and localization of abnormal video events. This architecture is illustrated from the previous architecture proposed in the Chapter3. Then we present the 3D CAE-based methods for the extraction of descriptive spatio-temporal representations. To enhance the architecture seen in this chapter, we introduce a new end-to-end architecture for deep one class classification for anomaly detection.

The Chapter5 concludes this thesis and discusses the future work.

Chapter 2

State of the art

2.1 Introduction

The rise in concern for security worldwide has led to the widespread use of video surveillance systems. The video stream generated is processing entirely by human operators which may make the surveillance task more and more difficult. Considering the issues surrounding this phenomenon, a real trend has been created around the development of intelligent video surveillance solutions. This concerted effort by the scientific and industrial community has resulted in the development or adaptation of a large amount of image processing approaches for video surveillance, among which we can cite the tracking methods adapted for traffic control, Person re-identification to check whether a person appearing in different images is the same person or not, Classification methods including the detection of abandoned luggage, event recognition, etc. Despite the positive impact of these approaches on the exploitation of surveillance camera, an important part of operators remains essentially unchanged, and the main objective of video surveillance operators is still to detect abnormal behavior that may represent security risks. In order to address this problem, the development of computer vision systems capable of training normal scene behaviors and detecting abnormal events has become critical. The automatizing detection of abnormal video events is an active research task in the computer vision community. Many works are regularly proposed in order to address this real problem. Abnormal video events have been called in literature by many names, such as anomaly, irregular behavior, uncommon behavior, unusual behavior , or abnormal behavior, ect, Popoola and Wang, 2012. These different names will be used alternatively without worrying about technical incoherence. The detection of abnormal video events is also characterized by a variety of strategies for dealing with training data. A first approach is to perform the training only on normal data and considers any type of events outside the training phase as abnormal. Another approach, in opposition to the first, is to use only abnormal events for training, Zhang et al., 2010. This approach can be effective in identifying a certain type of abnormal events, but has a high risk of missing abnormal events different from those trained. Another approach is based on the use of labeled data in two different classes, normal and abnormal, Zhou et al., 2016b. Other work uses more advanced classified and labeled data where each class represents a specific type of event, Lao, Han, and De With, 2009; Foroughi, Rezvanian, and Paziraee, 2008. Approaches that use abnormal events as learning data often have limitations. In the fact some abnormal events are impossible to reproduce. The variability of abnormal events significantly complicates the learning task and can negatively affect the modeling. As last and not least, generally based on clustering methods, it is to use unlabeled databases containing both normal and abnormal data, Roshtkhari and Levine, 2013,Javan Roshtkhari and Levine, 2013. It is assumed that normal events are those that occur frequently and abnormal events are those that occur rarely. The benefit of this approach is that it does not require any labelling of training data, but its effectiveness is compromised by the assumption that all rare events are abnormal because obviously a rare event is not necessarily abnormal. Despite the different strategies for training

data on abnormal event detection, Zhou et al., 2016b, Hasan et al., 2016, Lee, Kim, and Ro, 2018, Oza and Patel, 2019a. The first approach of using only normal data during the training has become the standard, Kiran, Thomas, and Parakkal, 2018. In the context of video surveillance, an abnormal event is often described as an event with a relatively low probability of occurrence in the monitored scene. In this context, the detection of abnormal events can be considered as a pattern and/or motion recognition task, under the constraint that the element to be recognized is not included in the training set. In our works for this thesis this approach was adopted.

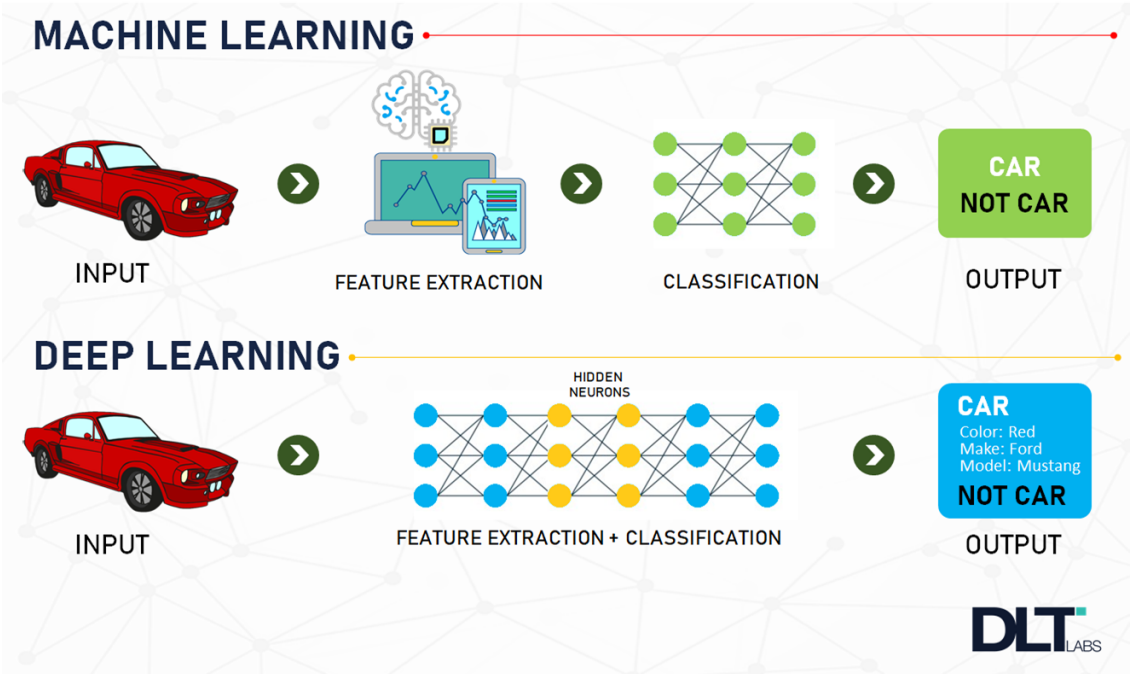


FIGURE 2.1: Recognition pattern models. On the top, the standard model based on targeted feature extraction enhanced by Machine Learning and on the bottom, the model based on deep learning.

Given the large amount of work done on the detection of abnormal events and their diversity, categorizing existing methods is not an easy task. In fact, the large majority of these methods have been proposed in the context of research work and each one makes a particular contribution, which makes it special and unique. However, in the following, we propose a first classification according to the adopted pattern recognition model. This classification is then refined by highlighting the similarities that may exist between different methods. The standard pattern recognition model essentially consists of three steps; data acquisition, obtaining new representations by extraction of representations (hand-crafted features) and the classification of these representations, Figure2.1. In the standard model, the classification step is done often by training classifiers while the feature extraction step requires manual processing to select and extract suitable features. In recent years, a second model has been imposed, thanks in particular to the emergence of deep learning when the step of extracting characteristics is replaced by a step of training representations, this technique is called "learning representations", LeCun, 2016 Figure 2.1. In this way the characteristics are automatically selected and extracted according to the specific task at hand. In the rest of this chapter we will present a detailed state of the art, including the most relevant methods for detecting abnormal video events.

2.2 Methods based on target feature extraction

2.2.1 Feature extraction

In general, the image is rarely exploitable by computer vision systems in its raw form. A treatment is often necessary in order to obtain a representation adapted to intended task(s). This representation is obtained by the extraction visual characteristics such as color, texture or gradient. Other characteristics such as optical flow or motion vectors, especially used in video processing, can exploit not only the image, but also its relationship with adjacent images in the time axis to extract information about motion. Many features have been collected and used in the field of abnormal event detection. In the literature these features are often split up in two categories according to the source of their extraction. A first category groups the features extracted at the pixel level, they are called "low-level features". A second category groups both the feature extracted at the level of the object and at pixel level, they are often designated in opposition to the first category by "high level features", Sodemann, Ross, and Borghetti, 2012, Popoola and Wang, 2012

2.2.2 Feature extraction at the object level

The most common feature extraction at object level is trajectory based methods, Piciarelli, Foresti, and Snidaro, 2005, Piciarelli and Foresti, 2006, Piciarelli, Micheloni, and Foresti, 2008, and Calderara et al., 2011, Jiang et al., 2011. These Methods exploit it generally trying to define a model for the normal trajectories of a scene and report any deviations from this model as abnormal. In certain application scenarios, the trajectory and position of the target objects are sufficient to detect any abnormal events ,Duque, Santos, and Cortez, 2007, road traffic monitoring, Dong et al., 2010, and counting people Biliotti, Antonini, and Thiran, 2005. The trajectory can be combined with other descriptors of the such as object size to get better representations. For example, in Zhang et al., 2010, the path, the distance between objects, the velocity of the objects and the energy of the movement have been combined to represent the events. The trajectory has also been combined with low-level features to detect not only abnormal behavior but also the complex behaviors related to finer movements. Despite of the proven efficiency of trajectory analysis in certain cases , but the complex behaviors related to more complex movements, it is not always possible to determine the exact trajectory of a movement. On the other hand, the large majority of methods that use trajectory require the application of precise tracking and object detection techniques, which makes them vulnerable to occlusions in crowded scenes. In addition their dependence on tracking and object detection algorithms are also characterized by high computational complexity. Other methods of extracting features less affected by occlusions have been used at the object level for the detection of abnormalities. In Tang, Wang, and Lu, 2009, Rectangles surrounding the objects as well as their widths and lengths were used as descriptors to detect abnormal behavior in elevators. Events can also be represented in the form of blobs. in Tao Xiang and Shaogang Gong, 2005, blobs are formed with the foreground pixels. The centers of these blobs and their sizes are then merged with other descriptors to obtain vectors of characteristics that are representative of the scene. The silhouette was also used in the recognition of video events. in Wang and Suter, 2007, extracts the silhouettes and transforms them through a dimensional reduction algorithm, in order to obtain exploitable representations for activity recognition.

2.2.3 Low-level features Extraction (pixel level)

Due to the challenges of video tracking and object detection in a CCTV scene, many abnormal event detection methods focus on extracting pixel-level features such as (texture,

gradient, and motion). Texture analysis returns information on the spatial arrangement of the pixel intensity of the pixels in the image. In ,Reddy, Sanderson, and Lovell, 2011, a filtering with a 2D Gabor wavelets Tai Sing Lee, 1996 is performed to obtain the texture and it is used to improve the dissociation between the different elements of the scene. Methods are also proposed to model simultaneously the appearance and dynamics of a scene using DT dynamic texture (dynamic texture), Mahadevan et al., 2010. The gradient has often been used in abnormal event detection, it is used to describe the appearance and local shape of objects in an image. The HOG (Histogram of Oriented Gradients) is one of the methods by which the gradient can be exploitedRoshtkhari and Levine, 2013, Zhao, Fei-Fei, and Xing, 2011. Moreover, The gradient can also be applied to the time domain by constructing a spatiotemporal gradient histogram, Zelnik-Manor and Irani, 2006, Li et al., 2015.

Considering the importance of movement in the event's characterization, the extraction of motion characteristics has been the topic of many methods, among which: MHI (motion history images), MEI (motion energy images), Bobick and Davis, 2001,Davis, 2001, pixel Change History, Xiang, Gong, and Parkinson, 2002, Tao Xiang and Shaogang Gong, 2005, However, the optical flow, Lucas and Kanade, 1981,**Twoframemotionestimation**has been the most widely used in the detection of abnormal events,Reddy, Sanderson, and Lovell, 2011,Feng, Zhang, and Hao, 2010,Sharif, Uyaver, and Djeraba, 2010,Wang and Snoussi, 2014. The optical flow can also be used in the context of histograms HOF (Histograms of Optical Flow) to describe the movement of a scene,Zhao, Fei-Fei, and Xing, 2011,Adam et al., 2008. Gradient, texture and motion used independently are generally not able to describe complex spatiotemporal events. However, these characteristics combined can be an effective description of the event, Zhao, Fei-Fei, and Xing, 2011,Reddy, Sanderson, and Lovell, 2011. Low-level feature extraction approaches have the advantage of being robust to occlusions that significantly affect tracking accuracy and do not require the prior use of object detection methods. However, the characteristics obtained with these methods are often criticized because of their lack of effectiveness in representing complex patterns in videos.

2.2.4 Classification and modeling

Classification/modeling is usually the next step after feature extraction in an abnormal event detection approaches. In this step, the representations obtained by feature extraction are exploited in order to dissociate between normal and abnormal events in a scene. In the literature many algorithms have been proposed to perform this task. SVM support vector machines (support vector machines) are one of the best and most popular classification methods. The SVM developed from the contributions of ,Vapnik, 1963,Vapnik, 2000,is a statistical learning method designed to find the optimal hyperplane separating two classes of data in a multidimensional space. It attempts to reach a compromise between minimizing experiential risks and preventing overffitting. SVM can also deal with non-linear classification problems using kernel methods,Boser, Guyon, and Vapnik, 1992,Piciarelli, Micheloni, and Foresti, 2008,,Wang and Snoussi, 2012,Bouindour et al., 2017. The SVM was initially adapted to the supervised classification of data into two classes. However, it has been improved to address a wide variety of classification problems involving multi-class classification Pittore, Basso, and Verri, 1999,Aggarwal, 2011, and one-class classification Schölkopf et al., 2001. The OC-SVM (One-Class SVM) can with data belonging to a one class (the positive class) and some outliers learn a discriminative boundary around the set of positive instances and detect the elements external to this set. The OC-SVM's capability to classify using mainly positive class data has led to its intensive use in the detection of abnormal events Schölkopf et al., 2001,Xu et al., 2017.

Clustering methods were also used to detect abnormal events. K -Means is an unsupervised data clustering algorithm that allows the samples of a data set to be grouped into K separate clusters. With this algorithm a sample is assigned to the cluster with the closest mean to it. In Tang, Wang, and Lu, 2009K -Means is used to associate labels to object representations, these labels are then exploited in order to build a model. [74] used K -medoids a variation of K-Means to detect abnormal trajectories. Other clustering methods inspired by the BOV (Bag Of Visual words) approach allow to represent the data through a dictionary, often called codebook in the literature on the detection of abnormal video events Javan Roshtkhari and Levine, 2013,Hasan et al., 2016,Li et al., 2015,Cheng, Chen, and Fang, 2015,Xiao, Zhang, and Zha, 2015. The codebook allows to represent all the data through codewords. These codewords are assigned to the different data samples thanks to a similarity measure. In Javan Roshtkhari and Levine, 2013, a video is divided into several spatiotemporal volumes thanks to the dense sampling. A codebook is constructed to represent these volumes using a Euclidean distance as a measure of similarity. Codewords are constructed by taking into consideration not only the samples it represents, but also their frequency of occurrence.

In addition, many others classification and clustering methods, model-based approaches have also been explored. The hidden Markov model (HMM) is one of the most intensively exploited methods for modeling behavior and detecting abnormal events Tang, Wang, and Lu, 2009,Utasi and Czúni, 2010,Zhang et al., 2010. The HMM is a graphical oriented model, it can be represented as nodes connected by transition links representing a time series of states. Each node represents a state that is not directly observable. However, for each state an observation corresponding to a set of state probabilities is performed. Two assumptions are imposed on HMM: 1) state transitions are only conditioned by the previous state , 2) The observations are conditioned only by the current state, therefore later observations are considered independent of each other in the current state. The HMM is defined by two matrices: the transition matrix, it corresponds to the transition probabilities between states and the emission matrix which contains the probabilities of observations. Both matrices can be determined by the Baum-Welch training algorithm. The success of HMM for modeling behavior and detecting abnormal events is most probably due to the time dependence associated with this method. As opposed to many other methods applied to the detection of anomalies, HMM is able to take into account the dynamic nature of the behaviour. Many variations of HMM have been applied for the detection of abnormal events. In Utasi and Czúni, 2010, an HMM and a mixture of Gaussians MOG (mixture of Gaussians) were used to detect abnormal events in road traffic areas according to the characteristics extracted using the optical flow.Zhang et al., 2010, used a CHMM (coupled hidden Markov model) to detect abnormal human interactions inside buildings. The CHMM is a model that makes two HMMs interact by adding transition probabilities between both of them. In particular, this model allows to model a stochastic process with more than one state at a given time, which can be useful to model not only the elements of a scene, but also their interactions. Another strategy is to arrange several models not in parallel, but in cascade, which allows the use of multiple different models, each of them sensitive to a specific type of event. Chung and Liu, 2008 presented a cascade structure named HC-HMM (Hierarchical Context Hidden Markov Model), it is composed of three modules to model events across three contexts: spatial, behavioral and temporal. The Markov random field (MRF) was also used to model the activity of a scene. The MRF is similar to the HMM in the sense that they are both graphical models used to model Markov systems. Meanwhile, Markov random fields are non-oriented graphical models. Kim and Grauman, 2009, an MRF has been proposed to detect abnormal activities in a video. The model is built using spatiotemporal regions of the video, each region is identified by a node and the neighboring nodes (neighboring regions) are connected by links. In

Benezeth, Jodoin, and Saligrama, 2011, a Markov random field model parameterized by a co-occurrence matrix was used to detect abnormal activities related to the direction, velocity and size of objects. Popoola, O. P., Wang, K. (2012). Video-based abnormal human behavior recognition—A review. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(6), 865-878.

2.2.5 Machine/Deep Learning based methods

For many years, the development of a pattern recognition system based on the traditional model required expertise and in-depth knowledge to extract from the raw data suitable representations that could be used to detect, identify or classify elements among the input data. Abnormal event detection methods that have adopted this model have the same dependencies. These methods require a priori knowledge to build a features extractor adapted to the targeted events and to the monitored scene. These constraints have led to the emergence of abnormal event detection methods based on learning representations and more precisely on deep learning. Representation learning or feature learning is a set of techniques to automate the feature extraction step. These methods allow to define, through learning, the adequate transformations to be applied to the input data in order to obtain representations to perform a targeted task such as action recognition, image classification, estimation of human pose, semantic segmentation, etc. Deep Learning is a sub-domain of machine learning, it aims to learn high-level abstractions in data using multi-level architectures. These different levels are obtained by stacking multiple modules of nonlinear transformations. Each module transforms the data at a different level until a suitable representation is obtained that makes it possible to carry out the target task. Deep learning has strongly contributed to the challenging of the relevance of the traditional model in some application cases, in the sense that it has made it possible to design efficient pattern recognition systems without in-depth expertise on the targeted elements.

Supervised Models

CNNs (Convolutional Neural Networks) are among the most popular supervised deep learning methods. This is largely the reason for the excellent results obtained with CNNs such as Alexnet, VGG, GoogLeNet and ResNet in international competitions such as ILSVRC Krizhevsky, Sutskever, and Hinton, 2017,Conneau et al., 2016,Szegedy et al., 2015,He et al., 2015, (ImageNet Large-Scale Visual Recognition Challenge) [84]. The CNN is a type of artificial neural network inspired from the animal visual cortex. It consists of several layers that process data in a hierarchical pattern, Figure 2.2

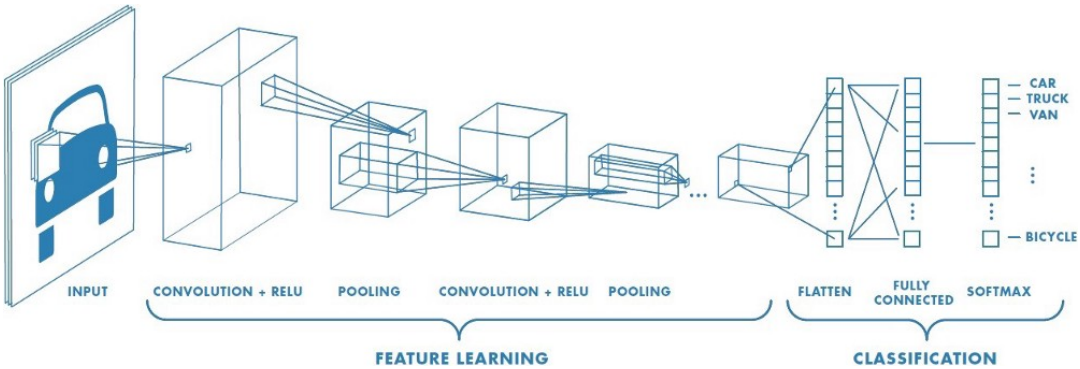


FIGURE 2.2: CNN

The features extracted, by CNN, in the first layers generally describe the presence of simple shapes (edges and contours), the following layers extract more complex patterns by detecting aggregations of simple shapes while overlooking irrelevant variations such as slight shifts or rotations of the patterns. The deeper more layers describe complex shapes, with an increasing level of abstraction, until they are able to represent parts of objects or even complete objects in the case of the last layers, Figure 2.3.

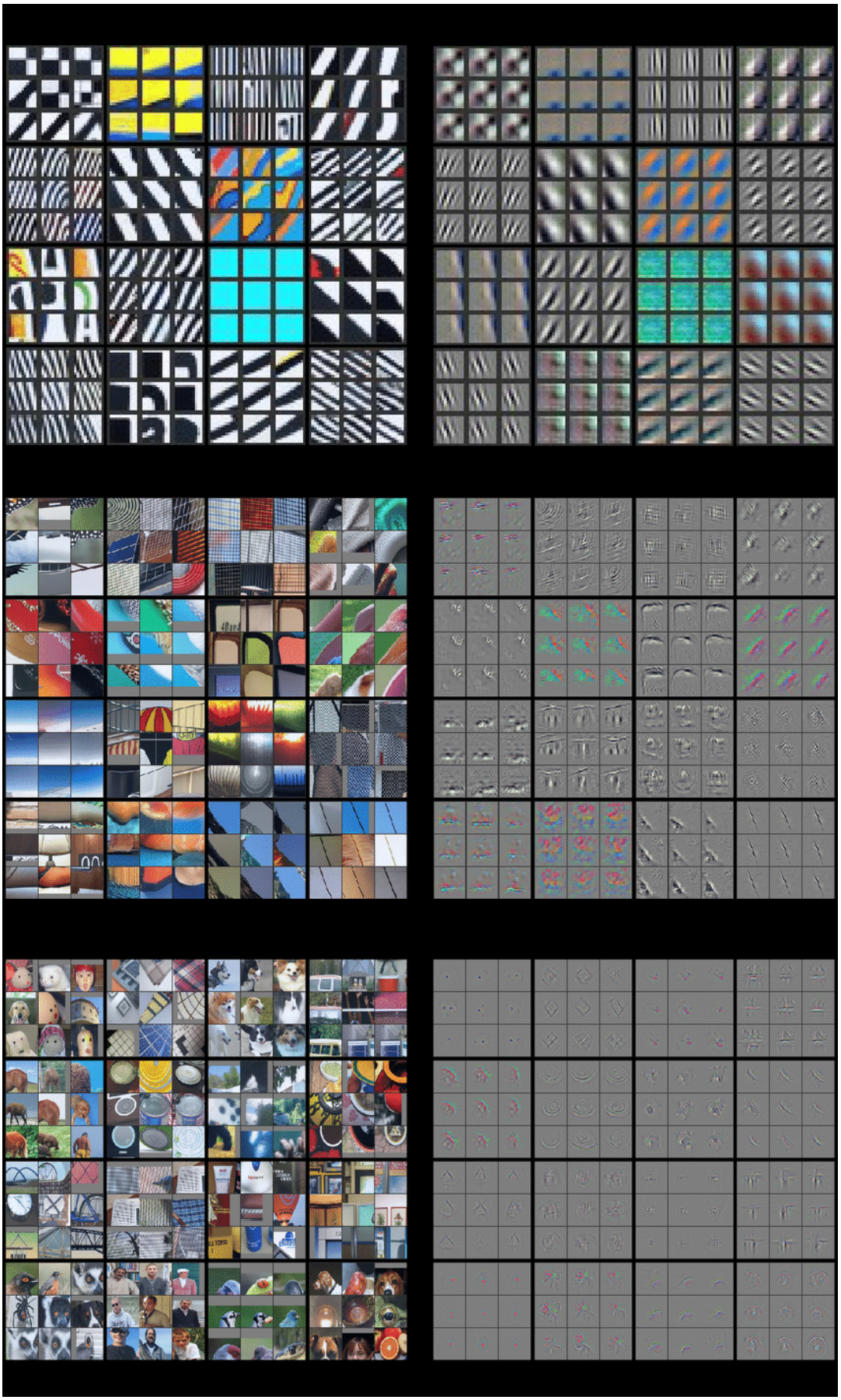


FIGURE 2.3: Visualization of feature activations at different layers in a CNN by a deconvolutional network (Zeiler and Fergus, 2014)

Each convolution layer consists of several units (neurons) distributed in the form of feature maps. A neuron inside a layer is connected to local regions, called receptive fields, in the feature maps of the previous layer. This connection is made using a set of weights called a filter. The CNN is also distinguished by what is called shared weights, all neurons in the same feature map share the same connection weight. This is done by applying the same convolution filter on the set of the previous layer. Weights imply that all neurons of the same map react to the same feature, but in a different way according to their respective field. In a CNN the convolution layers are usually associated with non-linear activation functions such as the ReLU (rectified linear unit). In addition to convolution layers with local connections and weight sharing, pooling is also a key word in CNNs. This operation takes the form of a rectangular sampling filter, applied to a local region of a feature map. An example of pooling is represented by the figure 2.4. Pooling ensures robustness to slight offsets and pattern distortions. It also makes it possible to reduce the size of the feature maps, which has the advantage of reducing the amount of network parameters and therefore makes it easier to learn and reduce its computational complexity. These different concepts (local connections, shared weights, pooling and deep architecture) have enabled CNN to overcome some of the limitations of its predecessors by better exploiting the powerful spatial and local correlation present in natural images. It is important to note that CNNs generally integrate a classification block consisting mainly of fully-connected layers, these layers are used at the end of a network after several layers of convolution and pooling in order to obtain a high-level abstraction. The CNN in its most common form is a supervised learning algorithm, its training requires a large amount of labeled data divided into several classes. Its training consists in empirically calculating the optimal values to be attributed to its different weights. The training is generally done using a gradient back-propagation algorithm. In the context of the detection of abnormal events, the CNNs have been exploited essentially according to two different approaches. the first one consists in training a CNN in a supervised way on a labelled image database. The second approach is based on the transfer learning, previously trained for other pattern recognition tasks.

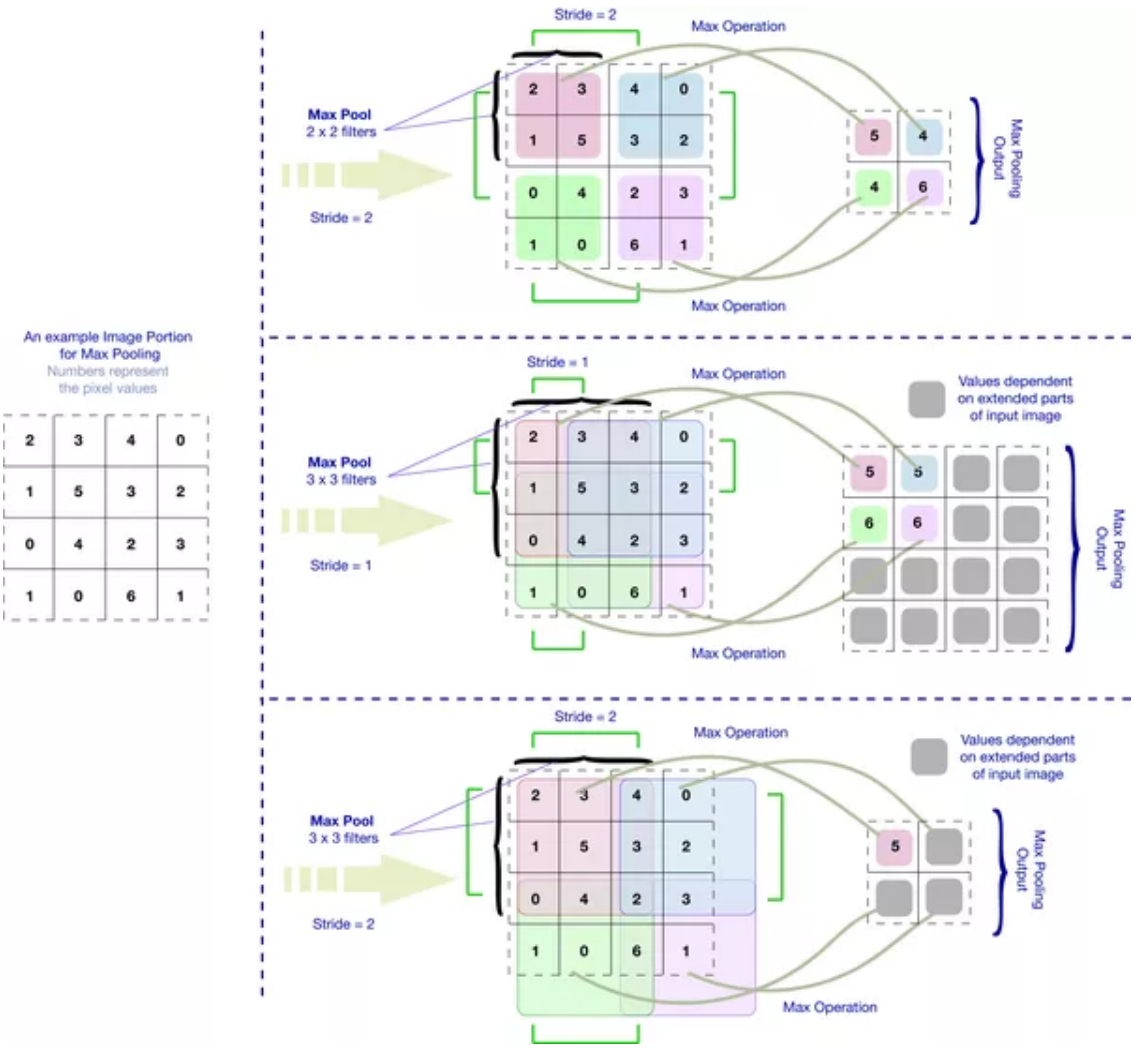


FIGURE 2.4: Example of polling layers

Supervised learning

As mentioned above, the CNN is a supervised learning models. To fully exploit these capabilities in terms of feature extraction and classification for anomaly detection, a labeled database containing examples learning of both classes (normal and abnormal) is required. In Ding et al., 2014, a 3D CNN is proposed to classify video clips into two classes (fighting or non-fighting) in order to detect acts of violence in ice hockey videos. A 3D CNN is defined by 3D convolution operations, which allows it to extract spatiotemporal features that are crucial information for motion description. In Zhou et al., 2016b, a 3D CNN was also designed to classify video volumes of interest SVOI (Spatial-temporal Volumes of Interest) in two classes: normal and abnormal. Volumes of interest are selected using the optical flow. [86] proposes to combine a multi-tasking Fast R-CNN with the kernel density estimation (KDE) method. The Fast R-CNN multi-task is trained in a supervised way to extract semantic features and classification scores for different objects in the input images. These features are then used by the KDE to detect anomalies. In this system we can not only detect abnormal events, but also provide a description of the detected event thanks to the labels. Two-stream architectures using CNNs have also been explored in the context of abnormal event detection. Jamadandi, Kottursheltar, and Mudenagudi, 2020 proposes to exploit dual networks to categorize images into two classes (normal and abnormal images). The first network is a pre-trained and fine-tuned CNN with trained images belonging to the two classes, it allows the

extraction of appearance-related representations. The second one is a CNN identical to the first one, but fine-tuned with optical flow representations extracted from image sequences. The last one in particular allows a better description of the movement. Once the networks are trained independently, the dissociation between normal and abnormal images is done by using the two classification scores returned by the two networks. Despite the successful results of methods based on supervised deep learning, the need to use both normal and abnormal training samples complicates their integration into intelligent video surveillance systems.

transfer Learning

It has been demonstrated that an CNN trained to perform a target task can provide generic and robust features that can be used to perform another computer vision task different from the one for which it was specifically trained. In Sharif Razavian et al., 2014, representations extracted with OverFeat, a CNN trained only for object classification, are exploited by linear SVM or a Euclidean standard for different tasks (scene classification, detailed classification, attribute detection, visual instance retrieval).The results provide tangible evidence on the ability of CNN to provide generic and robust features that can be used for different computer vision tasks. This principle has been applied in many works on the detection of abnormal events. In Bouindour et al., 2017, a 2D pre-trained CNN on image classification databases is modified to extract representations of the different regions of the input images. An OC-SVM is then used to detect among these regions the ones with abnormal events. Bouindour, 2019 combines a 3D CNN with an adaptive classifier similar to a codebook to detect abnormal events. The system can adapt to the appearance of new events through human interaction, which can prevent many false alarms. In Ravanbakhsh et al., 2016 a pre-trained CNN is combined with a binary quantization layer with weights trained by a binary hashing method called ITQ (Iterative Quantization Hashing) Gong et al., 2013. This network allows to obtain a measure of irregularity which is then combined with the optical flow to detect abnormal events. In Sabokrou et al., 2016, a pre-trained CNN is combined with a trainable sparse auto-encoder in order to obtain a two-level feature extractor. At the output of the CNN a first Gaussian classifier is used to classify image regions as normal, abnormal or suspicious. The representations of the suspicious regions are then transformed by the auto encoder to obtain more discriminant representations.

Methods based on learning transfer do not require a labeled database for feature extraction and their results in terms of detection and localization are very promising. However, the dependence of these methods on pre-trained models imposes a certain inflexibility considerably reduces their prospects for improvement. These criteria have encouraged the emergence of work focused on approaches based on unsupervised learning.

Unsupervised models

The development of learning methods that do not require labeled databases has always been a primary objective in the various fields of machine learning. In addition to the difficulty of creating tagged databases rich enough to capture the complexity of some of the topics. This interest in unsupervised learning is inspired in part by the fact that human learning is largely unsupervised LeCun, 2015. Indeed human have a considerable capacity to observe, analyze and understand the world around him without using labels for each situation. Despite the importance and challenges surrounding this type of learning, the rapid success of the CNN has for a time somewhat overshadowed unsupervised learning. Nevertheless, the increasing development of generative models in recent years has rekindled the interest of the scientific community in the development of methods based on unsupervised learning. This new interest

has been particularly useful in the field of abnormal event detection since many methods based on unsupervised learning have recently been explored.

Reconstruction Learning

Methods such as auto encoders (AEs) or sparse coding are used to extract different linear and non-linear representations of appearance (image) or motion (stream), in order to model normal behaviors in surveillance videos. The AutoEncoder (AE) is a fully connected neural network that is widely used in machine learning. It consists of an input layer, an output layer and one or more hidden layers, Figure 2.5. AE training is usually done by back-propagating the gradient in order to minimize the reconstruction error between input and output data. In AE the hidden layers are allocated between the encoder and the decoder, the encoder is used to encode the input data into a representation, generally more compact, the decoder is used to reconstruct the data according to the representation generated by the encoder. An AE is often used as an alternative to PCA for dimensional reduction and can also be an effective tool for feature extraction. Once trained, the encoder can be used to extract representations that can be used in different machine learning tasks such as clustering and outlier detection. Auto-encoder varieties such as the Denoising autoencoder DAE (Denoising autoencoder) and the Variational autoencoder VAE (Variational autoencoder) have broadened the application field of AE.

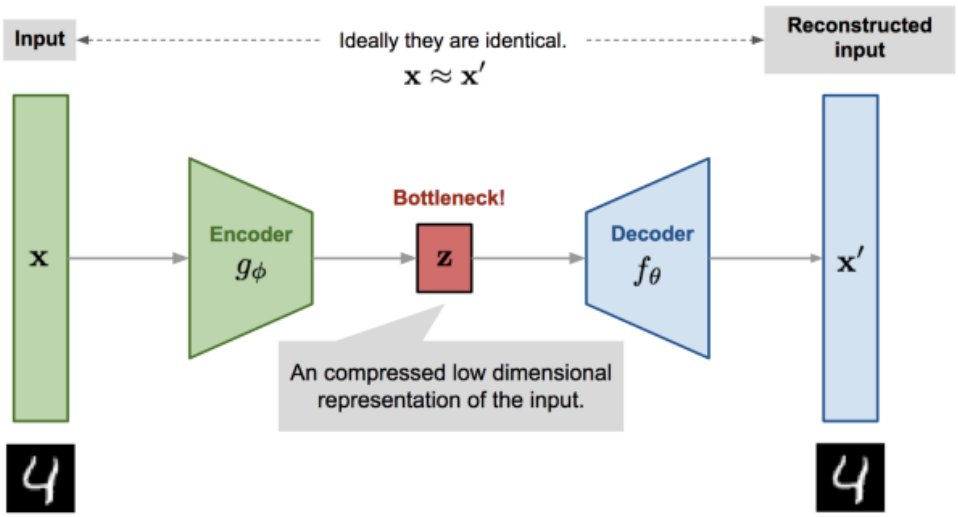


FIGURE 2.5: The building architecture for an auto-encoder

Considering its capacity for unsupervised learning, AE has been widely explored in abnormal event detection. [95] proposes AMDN (Appearance and Motion DeepNet) a network consisting of three SDAEs (stacked denoising autoencoders), a first one trained to reconstruct patches extracted from normal images, a second one trained with the optical flow representations corresponding to the patches and a third one trained with the concatenation of the patches and their optical flow representations. Once the three networks have been trained, the obtained representations are used to train three OC-SVMs. Thanks to this architecture, the detection of abnormal events is reduced to a binary categorization of the different image regions. Abnormal patches are detected through a combination of the decision scores of the three SVMs. Nevertheless, a derivative of the auto encoder called CAE (Convolutional AutoEncoder),Masci et al., 2011, using integrating deconvolution layers with shared weights. In

this way CAE preserves the existing spatial locality in real images. Gutoski et al., 2017 proposes to train a CAE for the reconstruction of 3D input volumes. Each volume consists of an image, the same image filtered by the Canny Edge Detector algorithm Canny, 1986, and the optical flow extracted from this image and its previous. After training the network, the training volumes (of the normal class) are reintroduced into the network. For each volume three reconstruction errors are obtained, one for each channel of the input volume. These three errors are combined under form of vectors and are used to train an OC-SVM. The detection of abnormal frames is then done by repeating the same error vector extraction procedure and using SVM to predict the class of each vector. In, Hasan et al., 2016, two methods also based on CAEs. In the first one, the authors suggest a CAE trained to reconstruct low-level features (HOG and HOF) extracted from normal class samples. In the second method, they propose to use a spatiotemporal CAE trained directly on video volumes. In both approaches, anomalies are detected using a regularity score calculated with the reconstruction error. In the same way,Chong and Tay, 2017b proposes to use the reconstruction error of a spatiotemporal CAE to detect abnormal events. The proposed CAE integrates 2D convolution layers for learning spatial features and ConvLSTMs (convolutional long short term memory) for temporal features. In ,Sabokrou et al., 2017 a cascade approach is proposed, as a first step, a two-layer auto-encoder is applied on video volumes in order to filter the normal and let the suspicious ones pass for further analysis. The second step consists of a CNN with weights obtained through unsupervised learning applied on AEs and then transferred to the CNN. In both steps, normal and abnormal volumes are differentiated thanks to multiple Gaussian classifiers placed in cascade and exploiting the hierarchical representations obtained with the different layers of AE and CNN. In addition to AE and its variants, other models based on a reconstruction approach have been explored. Zhou et al., 2019, proposes to exploit sparing coding for the detection of abnormal video events. The sparing coding applied to the detection of anomalies consists of two steps; a learning step to form a dictionary using the training data and a detection step, during which a sample is labeled as abnormal if its reconstruction based on the dictionary is impossible. The authors of,Zhou et al., 2019, present a new architecture called AnomalyNet, it is essentially composed of two networks, a feature extractor and an optimization network. Feature extraction is done through two steps, the first step consists in using a RankSVM ,Bilen et al., 2016 to compress a sequence of frames to obtain a single static image containing spatiotemporal information about the sequence. During the second step, these new images are processed by a pre-trained CNN to extract high-level features. The optimization network is an RNN (recurrent neural network) integrating SLSTM (Sparse LSTM) blocks derived from LSTM (long short term memory). Similar to the methods using AEs, abnormal events are identified thanks to the reconstruction error.

Predictive modeling

Another approach based on unsupervised deep learning tends to use predictive modeling for the detection of abnormal events. Differently from reconstructive models whose objective is to train a model to reconstruct the input data, predictive models try to predict a current sequence of frames using the previous sequences,Chong and Tay, 2017b, Medel and Savakis, 2016,Zhao et al., 2017, In other words, the objective is to model the conditional distribution $P(X_{t-1}/X_t)$, where X_t sequence of frames at the time t, X_{t-1} sequence of frames at the time $(t-1)$. AE has been widely exploited in this type of model. The function of an AE can be determined by considering its output values. When the output values are only the reconstruction of the inputs, the AE is a reconstructive model. When the output values are the values after the input values in the time axis, the model is said to be predictive. In, Medel and Savakis, 2016, a ConvLSTM-based AE is proposed for the detection of abnormal events, the network consists of: an encoder that extracts representations from an input sequence, a first decoder

that uses the representations extracted by the encoder to reconstruct the input sequence, and a second decoder that uses the representations to predict the next frame sequence. This architecture allows to obtain more robust representations, indeed the reconstruction branch generally only allows to learn representations to reflect the input data while the prediction branch allows to learn more temporal information to be able to predict the trajectories of the different objects in the scene. The network is only trained with normal frame sequences, which allows a higher reconstruction error when a sequence containing an abnormal event is introduced. Similarly ,Zhao et al., 2017, proposes a network made up of an encoder and two decoders, the first for reconstruction and the second for prediction. In this network, 3D convolution layers are used instead of ConvLSTMs, for the learning of spatiotemporal representations. In this network, 3D convolution layers are used instead of ConvLSTMs to learn spatiotemporal representations. In a 2D convolution layer the convolution is applied only to the spatial dimensions, whereas in a 3D convolution layer in addition to being applied to the spatial dimensions, the convolution is also applied to the temporal dimension, which allows to obtain spatiotemporal representations describing both shapes and movements present in the input sequence.

Generative models

In recent years, the use of Generative Adversarial Networks (GANs) has increased considerably in the fields of machine learning. GAN is an unsupervised learning algorithm first proposed by Goodfellow et al., 2014. It consists of two sub-networks, a generator and a discriminator in competition, Figure 2.6. During the learning phase the generator try to generate convincing data to fool the discriminator who in turn tries to detect whether the data is real or generated. In this way we obtain two trained networks, one to generate realistic data and the other to distinguish between real data and generated data.

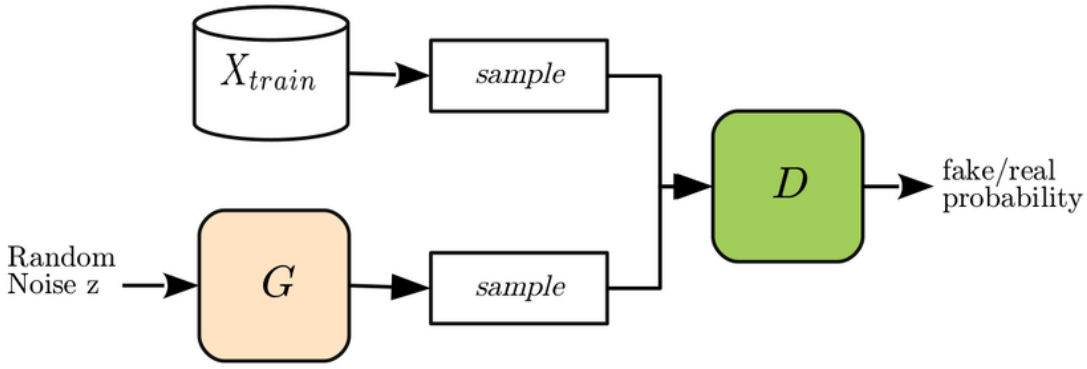


FIGURE 2.6: Generative Adversarial Network (GAN)

After the training phase, the generator can be used independently to create data,Isola et al., 2016,Jin et al., 2017 or for discrimination tasks,Ravanbakhsh et al., 2017,Liu et al., 2018, but it can also be used in conjunction with the discriminator,Shi et al., 2015,Sabokrou et al., 2018a. The ability to generate GANs has generated a lot of interest and many applications integrating GANs have emerged in various fields, among the most unexpected applications we can cite: Obvious, Sabokrou et al., 2018b a French collective of artists who have distinguished themselves by the creation of artistic paintings thanks to GANs. The generation of music, Lee et al., 2017 or the creation of cartoon personalities, Jin et al., 2017. However, GANs have also been strongly exploited in more "conventional" machine learning applications such as object detection, Ehsani, Mottaghi, and Farhadi, 2017,Li et al., 2017,

object transfigurationZhou et al., 2017 or the creatio6 of high-resolution images from low-resolution images Ledig et al., 2016.The popularity of GAN, has led many researchers to use it for anomaly detection. In Lee, Kim, and Ro, 2018, STAN (spatio-temporal adversarial networks) a generative adversarial network GAN (Generative Adversarial Networks) is proposed to meet the challenge of video anomaly detection. It consists of two sub-networks, a generator composed of convolution layers, ConvLSTM ,Shi et al., 2015, and deconvolution layers. In addition to the generator, the network also contains a discriminator composed of 3D convolution layers. The two sub-networks are trained in an adversarial process, the generator tries to produce images based on a sequence, while its competitor, the discriminator tries to detect whether a sample is real or the output of the generator. Once the two networks are trained with normal images, the detection of abnormal events can be done directly by the discriminator, as the discriminator has been trained only to accept normal sequences (containing no abnormal events) as real. Detection can also be done using only the generator, using the error with which the images are generated. However, the best results in, Lee, Kim, and Ro, 2018,are obtained by combining the decisions of the two networks. The author of Ravanbakhsh et al., 2017 also proposes the use of GANs for the detection of abnormal events. A thresholding of the generation error of the two GANs is used in order to identify the image regions containing the abnormal events. The first GAN is trained to generate optical flow representations from images and the second GAN is trained to generate images from optical flow representations. However, the error between the generated images and the real images is not sufficient to obtain convincing results. The author then uses a pre-trained CNN to extract new representations from the original images and the generated images and then calculates the error between these representations. This error is finally combined with that of the optical flow, which was initially usable to detect abnormal regions in the input images. Sabokrou et al., 2018a, proposes a method called AVID (Adversarial Visual Irregularity Detection) to detect and locate irregularities in videos. A GAN composed of a generator trained to erase the irregularities of the input images and replace them by the dominant patterns of these same images and a discriminator, an FCN (fully convolutional network) to predict the probability of different regions (patches) of the input images. Both networks are adversely trained and irregularities are simulated using Gaussian noise. After the training phase, each of the two networks is able to detect irregularities : the generator at the pixel level through the error between the original images and the generated ones, the generator has been trained to erase irregularities, therefore, when an image containing irregularities is introduced, the generator eliminates these irregularities and replaces them with other patterns which will result in a higher generation error. The discriminator, on the other hand, can directly predict the probability of a patch containing irregularities. However, in Sabokrou et al., 2018a, it has been shown that detection is more accurate by crossing the results of the two networks. In Wang et al., 2018 a cascade approach is proposed for the detection of abnormal events. The first step of this approach consists of detecting and extracting the foreground of the different images using a FCN. Optical flow representations relative to the foreground objects are then extracted. A first shallow network based on a variational autocoder is used to filter the image regions whose normality is obvious. Suspicious regions and their representations in terms of optical flow are then analyzed by a second, deeper network. Both networks are trained to reconstruct not only the images (foregrounds of the images), but also the optical flow representations. Thanks to a thresholding of the reconstruction error, this permits to isolate the anomalies both on the images, but as well on the optical flow representations.

One-Class models

Abnormal event detection approaches based on reconstructive, predictive or generative models are generally based on the assumption that a model formed on normal images will not

be able to reconstruct, predict or generate abnormal images. Therefore, a thresholding of the reconstruction, prediction or degeneration error is often used to detect abnormal events. However, in the case of video events, the different elements of normal and abnormal situations are often similar and it is usually their interactions or the context that defines the normality or abnormality of a situation. Discriminative models, Sabokrou et al., 2017,may be an alternative less impacted by this p,henomenon. In this respect, recent work aimed to develop one-class networks was proposed. Sun, Shao, and He, 2019 proposes DOC (Deep One-Class), an end-to-end trainable convolutional neural network, using only learning examples from an unique class. The network is obtained by replacing the softmax usually used in CNNs by an OC-SVM. Concretely, after introducing the input data into the network, the CNN standard layers extract representations relating to these data and the last layer (the OC-SVM) is responsible for defining the hyperplane that can separate these representations from the origin with a maximum margin. The network used consists of two 2D convolution layers, two pooling layers, a fully connected layer and a last layer to integrate the OC-SVM. The authors define an objective function that allows the training not only of the OC-SVM layer, but also of all the trainable layers of the network. In this way the network is optimized to extract compact representations and define the appropriate hyperplane to isolate the data representations from the target class. Further work on one-class neural networks has been proposed for anomaly detection,Chalapathy, Menon, and Chawla, 2018 ,Ruff et al., 2018,Perera and Patel, 2019. these works require very little adaptation to be exploitable in the context of detection of abnormal video events. In Chalapathy, Menon, and Chawla, 2018 a fully connected neural network is proposed for the anomaly detection. The network consists of a unique fully connected layer taking as input a data vector and returning as output a scalar. The network is trained thanks to an objective function inspired by the OC-SVM, The network is not used with raw data, but with representations extracted through an auto-encoder. It exploits and redefines these representations in order to optimize them to create an adapted boundary (hyperplane) that isolates normal data from anomalies. In the same perspective, Ruff et al., 2018 proposes to exploit the SVDD (Support Vector Data Description) in a neural network. The SVDD is similar to the OC-SVM in the concept that it is used to create an optimal boundary by separating the representations of the target class and dissociating them from the outliers.The objective of the SVDD is to define the most compact hypersphere capable of encompassing the majority of the target class representations. To effectively meet the challenge of deep learning in deep one class. Perera and Patel, 2019 proposes the use of the transfer learning for adapting pre-trained networks to anomaly detection. The authors assume that the two important aspects, compactness and description of the extracted features by a deep network, must be imperatively integratable.the description provides descriptive features. However, the compactness is used in order to ensure that images of the same class are described by similar representations, so they are positioned compactly in the characteristic space. These two aspects can be very significant contribution to decrease the intra-class distance and increase the inter-class distance. To obtain these two aspects, the authors propose two networks : a reference network R and a secondary network S. They also propose the use of two loss functions : a compactness loss integrated at the secondary network output and descriptiveness loss integrated at the R network output. The two networks are in parallel and sharing the same weights during the training. It is important to note that this architecture requires two different datasets: a first target dataset containing one class (the target class) and a second reference dataset containing several image classes. Moreover, The images as well as the classes of the reference dataset must not be used for training of first network. After the learning, the two identical networks capable of providing both described and compacted representations. These networks can be applied with a One-Class classifier to dissociate the elements of a target class from the outliers. Oza and Patel, 2019b introduces an architecture provides both features extraction and classification. The feature extractor is a pre-trained

CNN used to extract representations from the images of the target class. The classifier is a fully connected neural network to dissociate representations into two distinct classes (positive and negative class). Given the lack of availability of learning samples related to the positive class, the authors propose an artificial generation of data integrated into the network to replace the representations of the positive class.

2.2.6 Conclusion

A method for detection of abnormal video events generally consists of two essential steps: the first is the extraction of descriptive representations of the events in the scene. The second step is performed after the representations have been obtained and allows the detection of anomalies through their classification. In the literature two fundamentally different strategies are proposed. The first is based on the standard pattern recognition model and relies on targeted feature extraction. While a second, more recent approach, based on deep learning, for describing events. In this chapter, we have provided an overview of the main approaches based on these two strategies. We have taken into consideration not only theoretical consistency, but also the prospects for further development and concrete possibilities for integrating these methods into intelligent monitoring system. We have set the advantages and constraints of the different approaches listed in order to highlight the research approaches that are most likely to provide concrete results.

Chapter 3

Transfer and Unsupervised learning for anomaly detection

3.0.1 Introduction

Recently, deep convolutional neural networks have emerged as a powerful learning tools, specifically adapted to large amounts of data. these networks have demonstrated their superiority over standard methods, mainly based on targeted feature extraction, by improving the results established in many pattern recognition tasks such as object detection and localization Sermanet et al., 2013, Krizhevsky, Sutskever, and Hinton, 2017, video classification Karpathy et al., 2014, segmentation Long, Shelhamer, and Darrell, 2015, etc. Deep neural networks are not only able to achieve successful and positive results in many of the learning tasks for which they have been trained, but also provide usable representations for various pattern recognition tasks. Many works has exploited this potential use of networks trained to classify objects on large datasets such as ImageNet Krizhevsky, Sutskever, and Hinton, 2017 to perform other recognition tasks, where training data are less available Andrews et al., 2016. In particular, it has been shown that representations extracted with the use of a CNN, which are used only for object classification, could be exploited by standard classifiers such as the SVM in various tasks other than object classification Donahue et al., 2014,Nanni, Ghidoni, and Brahnam, 2017. This chapter proposes, encouraged by these results, to study transfer learning (Figure 3.1) in the context of abnormal event detection. In this regard we propose two methods based on pre-trained CNNs. First we propose to use pre-trained CNN (convolution neural network) and HOF (Histogram of Optical Flow) features for spatio-temporal feature extraction at frame level. Second, we try to fine-tune a pre-trained CNNs to construct CAEs for abnormal events detection using the distance of Mahalanobis. In the context of convolutional neural networks, the use of a CNN previously trained to solve a given task in an other given new learning task, without necessarily being related to the first one, is commonly referred to as "learning transfer". This designation will be adopted in this chapter and the rest of the thesis. In the next subsections we will first start with a review about the main component of the first methods such as VGG16, HOF (Histogram of optical flow), One-Class SVM. Then, We will introduce our first method and explain its different steps, before concluding this section with comparison results. The next section discusses the second method and its components such as Mahalanobis distance classifier and deep one class classifier. The third and final section will conclude this chapter.

3.1 Hybrid transfer learning and handcrafted features extraction

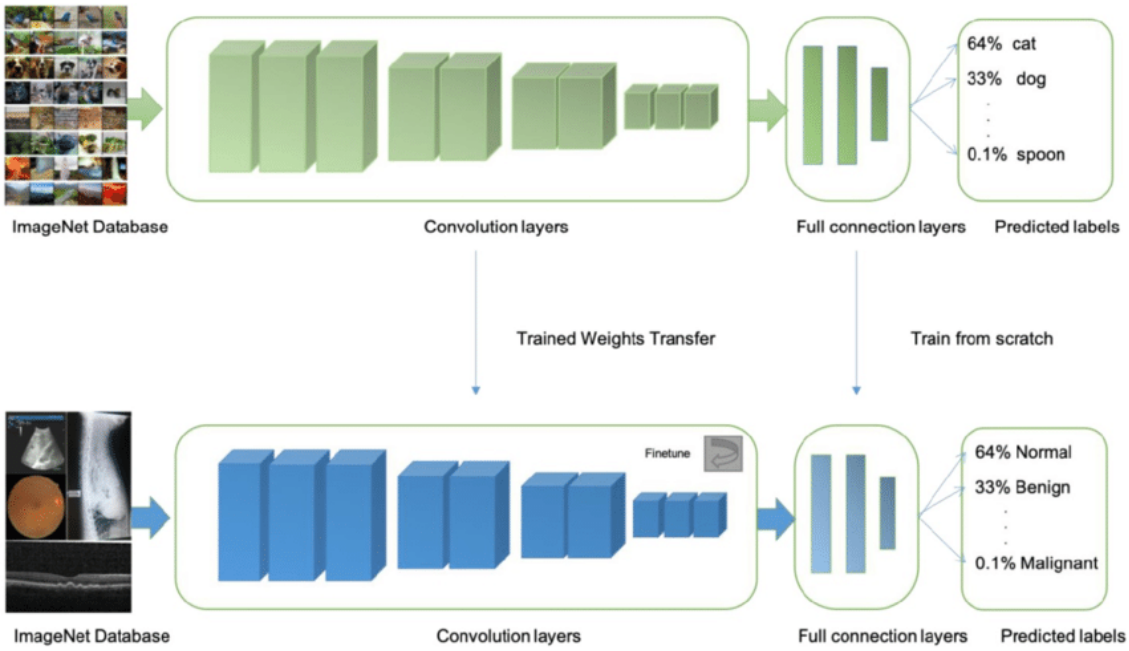


FIGURE 3.1: Illustrations of transfer learning: a neural network is pretrained on ImageNet and subsequently trained on retinal, OCT, X-ray images, B-scans for different disease classifications, Xu, Xue, and Zhang, 2019

3.1.1 VGG16

The VGG16, Simonyan and Zisserman, 2014 is a convolutional neural network model suggested by K. Simonyan and A. Zisserman of Oxford University in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition". This model has an accuracy of 92.7% in the top 5 tests of ImageNet, a data set of more than 14 million images belonging to 1000 classes. It is one of the successful models presented at the ILSVRC-2014. It provides the improvement over AlexNet by replacing kernel sized filters (11 and 5 in the first and second convolutional layer, respectively) with several kernel sized filters 3×3 one after the other. The VGG16 was trained for several weeks and was using NVIDIA Titan Black GPUs.

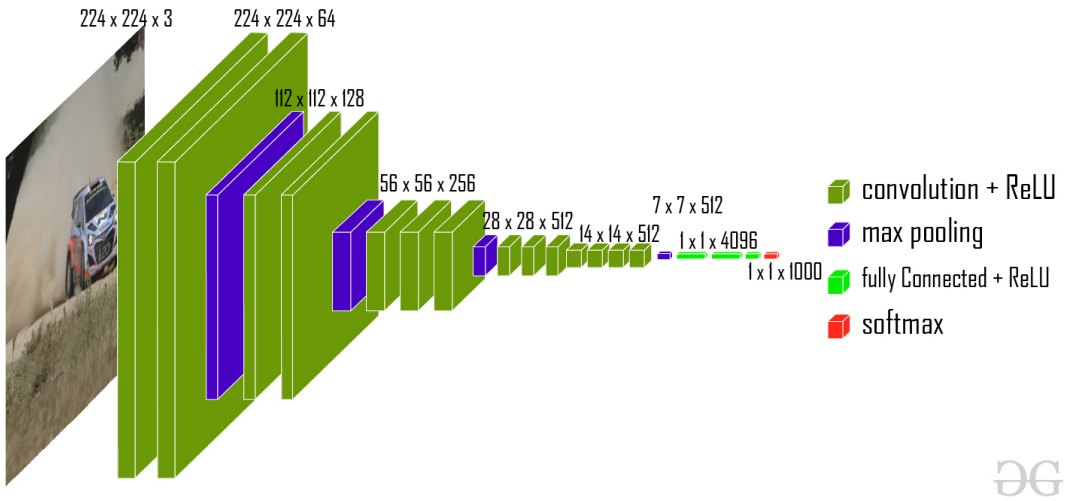


FIGURE 3.2: VGG16 Configuration

DataSet

ImageNet, Deng et al., 2009 is a dataset of over 15 million labeled high-resolution images belonging to roughly 22,000 categories. The images were collected from the web and labeled by human labelers using Amazon's Mechanical Turk crowd-sourcing tool. Starting in 2010, as part of the Pascal Visual Object Challenge, an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) has been held. ILSVRC uses a subset of ImageNet with roughly 1000 images in each of 1000 categories. In all, there are roughly 1.2 million training images, 50,000 validation images, and 150,000 testing images. ImageNet consists of variable-resolution images. Therefore, the images have been down-sampled to a fixed resolution of 256×256. Given a rectangular image, the image is rescaled and cropped out the central 256×256 patch from the resulting image.

Architecture

The input to first conv net layer is of fixed size 224 x 224 RGB image. The image is passed through a stack of convolutional (conv.) layers, where the filters were used with a very small receptive field: 3×3 (which is the smallest size to capture the notion of left/right, up/down, center) Figure 3.2. So we can conclude that VGG 16 is a CNN for general purpose and adapted to do transfer learning. In one of the configurations, it also utilizes 1×1 convolution filters, which can be seen as a linear transformation of the input channels (followed by non-linearity). The convolution stride is fixed to 1 pixel; the spatial padding of conv. layer input is such that the spatial resolution is preserved after convolution, i.e. the padding is 1-pixel for 3×3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling is performed over a 2×2 pixel window, with stride 2.

Three Fully-Connected (FC) layers follow a stack of convolutional layers (which has a different depth in different architectures): the first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains 1000 channels (one for each class). The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all networks. All hidden layers are equipped with the rectification (ReLU) non-linearity. It is also noted that none of the networks (except for one) contain Local Response Normalisation (LRN), such normalization does not improve the performance on the ILSVRC

dataset, but leads to increased memory consumption and computation time.

VGG16 significantly outperforms the previous generation of models in the ILSVRC-2012 and ILSVRC-2013 competitions. The VGG16 result is also competing for the classification task winner (GoogLeNet with 6.7% error) and substantially outperforms the ILSVRC-2013 winning submission Clarifai, which achieved 11.2% with external training data and 11.7% without it. Concerning the single-net performance, VGG16 architecture achieves the best result (7.0% test error), outperforming a single GoogLeNet by 0.9%, It was demonstrated that the representation depth is beneficial for the classification accuracy, and that state-of-the-art performance on the ImageNet challenge dataset can be achieved using a conventional ConvNet architecture with substantially increased depth Zhang et al., 2015.

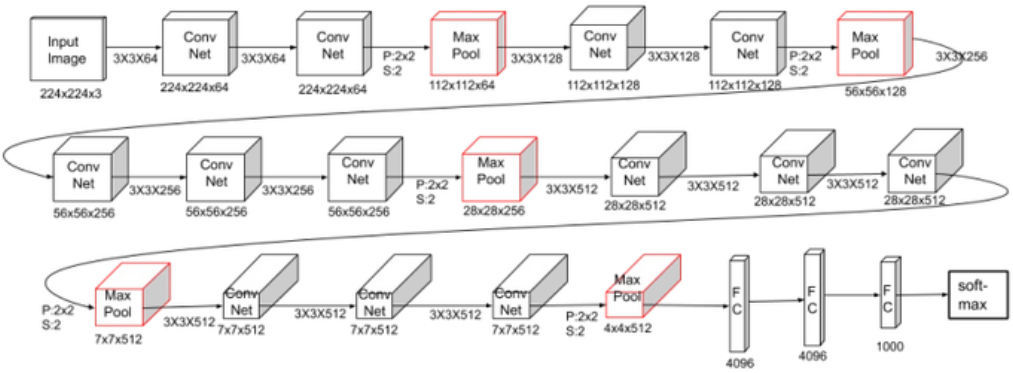


FIGURE 3.3: VGG16 Architecture

In our work, we propose to use the fully connected layers, these layers hold a reduced and a deeper representation of Feature Maps (FMs) generated by the convolutions layers to describe the whole frame. Therefore, we extract a deep features from the two last fully connected layers FC1 and FC2 of VGG16 ConvNet contain 4096 neurons to decide if the frame contained abnormal shapes or not (figure 3.4).

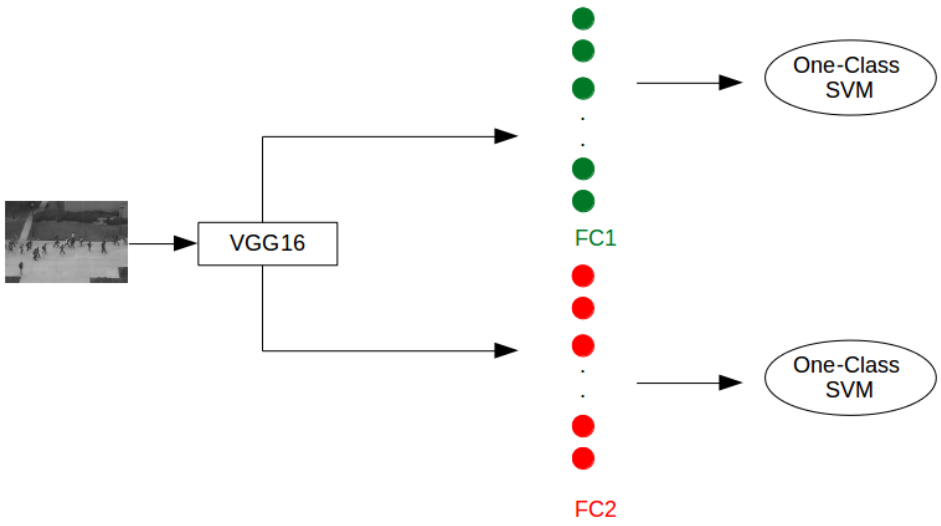


FIGURE 3.4: Feature extraction of shapes

3.1.2 Histogram of optical flow

Optical flow orientations have been used in various works. Dalal, Triggs, and Schmid, 2006, Dalal, 2006, In particularly, an histogram optical flow is method that has been applied to different tasks such human body identification when derivatives of optical flow, du and dv, have been taken in account in this work. In Utasi and Czúni, 2010, a histogram of optical flow orientations in the region of interest (ROI) was applied to build a model, the magnitude of the optical flow vectors was overlooked. While in our work, only the magnitude of optical flow is considered as the weight to calculate the histogram. In Adam et al., 2008,Kwak and Byun, 2011], optical flow was used as the basic feature to characterize behavior. The frame was split into small patches, and a bag-of-words feature was computed to represent the patch. In our work, the histograms of optical flow (HOF) descriptor is computed over dense grids of overlapping blocks. In Laptev et al., 2008, a histogram of optical flow was computed in the neighborhood of detected points to build a spatio-temporal descriptor. A weighted vote of each pixel is calculated for the edge orientation histogram channel based on the optical flow element orientation centered on it, then the votes are gathered into orientation bins over local spatial regions. The optical flow magnitude of a pixel is considered as a weight in the voting process. However, In our work, the VGG16 CNN does not take into account the motion concept. That is why, we propose handcrafted method to extract feature motion to pick up any abnormal movement into video in the following way: First we calculate a dense Optical flow (OF) (3.1) with Farnebäck, 2003 algorithm at every frame of the video describing the motion at each pixel. Each flow vector is composed according to its magnitude R and primary angle from the horizontal axis. Then, we establish the histogram of quantity motion depending on the magnitude R (3.2) only with two bins (3.3) the first bin represents the Low Motion Level (LML) and the second contains information about the High Motion Level (HML):

$$u = dx/dt, v = dy/dt \tag{3.1}$$

Where dx and dy represent the motion taken after dt time from frame to another. u and v respectively represent horizontal and vertical movements between two consecutive images.

$$R = \sqrt{u^2 + v^2} \tag{3.2}$$

$$F_i \longmapsto HOF = [B_0, B_1]_{i \in [|1,n|]} \tag{3.3}$$

F_i : The ith frame in video (i>0)
B_0 : Low motion, first component of HOF
B_1 : High motion, second component of HOF

3.1.3 One-Class SVM

The OC-SVM (One Class SVM) is an extension of SVM for one-class problems [72]. Widely used for anomaly detection, the OC-SVM allows to learn a hyperplane to separate the target class data from the origin in a Reproducing Kernel Hilbert Space (RKHS). The objective of SVM is not only to find the separator hyperplane, but also to maximize the distance between the hyperplane and the origin of space, figure 3.5. The data projection to the RKHS is done through kernel functions that allow to transform non-linear problems into linear problems. The RKHS where the data is projected is often referred as feature space and projections are referred as feature vector.

Optimization objective of the OC-SVM

Given a set of training data belonging to the target class, $X = \{x_i, i = 1,..,..,n\}$, $x_i \in R^d$, and , $\Phi : X \rightarrow H$ a projection function from the data space to the feature space H, the objective of the OC-SVM is to isolate the data projections with a decision hyperplane with maximum distance $\frac{\rho}{\|w\|}$ to the origin. Maximizing this distance is equivalent to minimizing $\|w\|$ and $-\rho$, The objective of the OC-SVM can then be formulated as a constrained optimization problem :

$$\min_{w,\rho,\xi} \frac{1}{2}\|w\|^2 - \rho + \frac{1}{\alpha_n}\sum_{i=0}^{n}\xi_i \tag{3.4}$$

under the constraints :
$< w, \phi(x_i) > \geqslant \rho - \xi_i, \xi_i > 0$

where the x_i are the training samples and the ξ_i release variables of the constraints, introduced to allow flexibility in the optimization problem, $\|.\|$ is the Euclidean distance and <.> is the dot product. The expression α_n introduced to manage the trade-off between maximizing the hyperplane distance from the origin and minimizing errors. Such as n is the number of training samples and $\alpha \in$ [0, 1] a specific OC-SVM parameter. It defines an upper limit on the ratio of outliers and a lower limit on the number of training samples used as support vectors. Given the importance of this parameter, the OC-SVM is often referred in the literature as α-SVM, the equation $< w, \phi(x_i) > -\rho = 0$ defines the decision hyperplane, such that w is the normal to this hyperplane and ρ is a bias. The projection function ϕ allows to solve a non-linear classification problem by constructing a linear classifier in the feature space H. By applying the Lagrange multiplier method, the dual problem of the OC-SVM is expressed as follows :

$$\min_{\beta} \frac{1}{2}\sum_{i=0}^{n}\sum_{j=0}^{n}\beta_i\beta_j K(x_i,x_j) \tag{3.5}$$

under the constraints :
$0 \leqslant \beta_i \leqslant \frac{1}{\alpha_n}, \sum_{i=0}^{n}\beta_i = 1$

When β_i are the Lagrange multipliers coefficients and $K(x_i,x_j) =< \phi(x_i), \phi(x_j) >$
By introducing these multipliers the decision function is defined as follows:

$$f(x) = sign(\sum_{i=0}^{n}\beta_i K(x_i,x) - \rho) \tag{3.6}$$

When $f(x) = 0$ the x sample is classified as normal, otherwise it is considered abnormal (outlier).

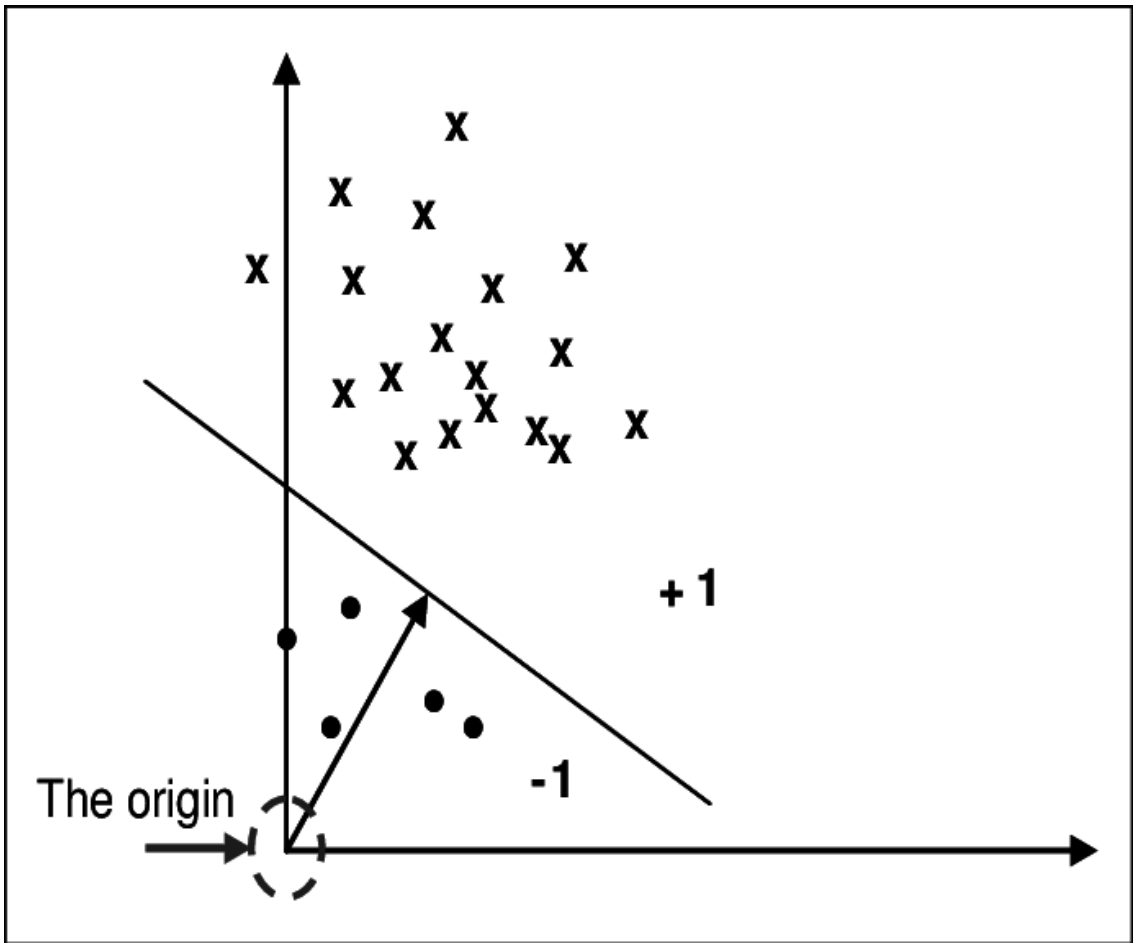


FIGURE 3.5: One-Class SVM

Kernel functions

For basic classification problems with small dimensions, data from different classes can be separated using linear separators (e.g. straight line or plane). However, many classification problems are not linearly separable. For this type of problems, a solution is to use a kernel function to project the data to a higher dimensional space where a linear solution would exist. Some of the most commonly used kernel functions in the context of statistical learning include the following :

Linear kernel : $K(x,y) = x \cdot y$

The linear kernel is the simplest kernel function, it is given by the usual scalar product. Algorithms using this function are often equivalent to their non-kernel counterparts.

Polynomial kernel : $K(x,y) = (x \cdot y + 1)^p$

This kernel allows to transform linear algorithms into polynomial algorithms. It allows to examine not only the features of the input samples to determine their similarity, but also the combinations of these features.

Gaussian RBF Kernel : $K(x,y) = \exp(-\frac{\|x-y\|^2}{2\sigma^2})$

The radial basis function (RBF) allows to apply a Gaussian scale on the distance between the training samples, which generates a projection space of infinite dimension.

Sigmoid Kernel : $K(x,y) = \tanh(\gamma \cdot x \cdot y + 1)$ The sigmoid kernel also called hyperbolic tangent kernel is derived from the Neural networks where the sigmoid function is often used as an activation function.

3.1.4 Classification

In this section, we explain how we can classify our features in order to determine the abnormal images. One-Class SVM (Support Vector Machine) is used to classify our feature shapes within unsupervised learning mode, which is coherent with our problem as we use only the normal event for training task. Moreover, a classification by thresholding is used to pick any anomalies in motions.

Support Vector Machine is statistical learning method adapted to non-linear problems with using kernel methods **c20** for both regression and classification problems (4):

$$N(X,X') = \Theta(X).\Theta(X') \tag{3.7}$$

Where $\Theta(X)$ is the projection of input data X to the new space H where the problem have a linear solution.

The target of this learning method is to find a separator "hyper-plane" to classify our data (5). The optimal classifier can be determined by maximizing the margin and it is represented by minimization problem shown in (6).

$$f(X) = W.X + b \tag{3.8}$$

$$Min\frac{1}{2}||W||^2 \tag{3.9}$$

subject to:

$$Y_i \times (W.X_i + b) \geq 1, i \in [|1,n|] \tag{3.10}$$

where n is the size of input training data and Y_i is data label (-1 or +1) defined by (8).

$$y(X) = sign(f(X)) \tag{3.11}$$

Moreover, in one-class SVM, the data from only one class are available which is consistent with our problem framework as only the normal event examples should be used for training. On another hand, a classification by thresholding (Fig. 2) is proposed to determine abnormal motions. As we explain in the previous section, the motion information at each frame is represented by a HOF with 2 bins, the first describes the low motion and the second describes the high motion (3). We fix the threshold 'S' as the maximum high motion in the training phase (9). Then, we compare the high motion B'_{1i} of each new frame F'_i with the threshold 'S' to decide if the frame contained abnormal motion or not figure 3.5.

$$S = Max(B_{1i}, B_{12}, ..., B_{1i}, B_{1n}) \tag{3.12}$$

B_{1i} : High motion in ith frame at Train phase

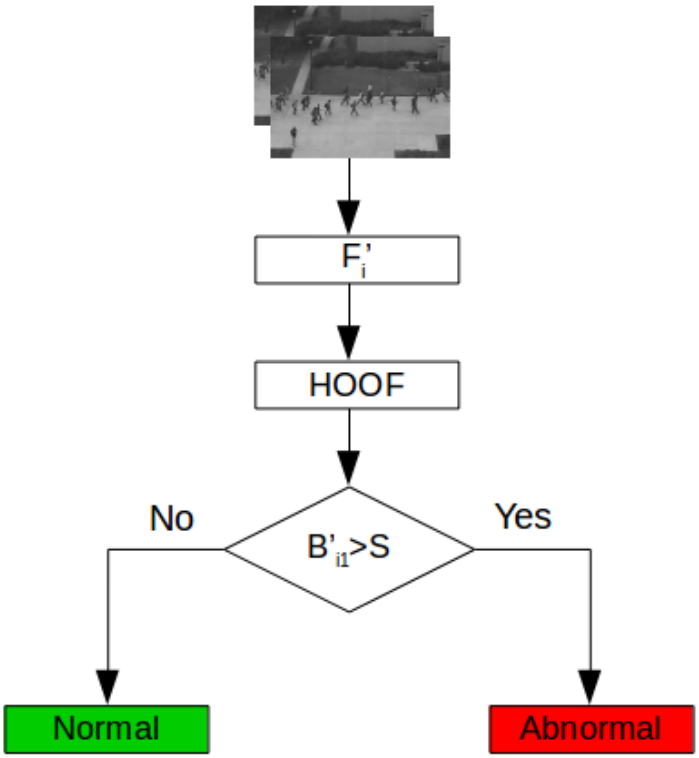


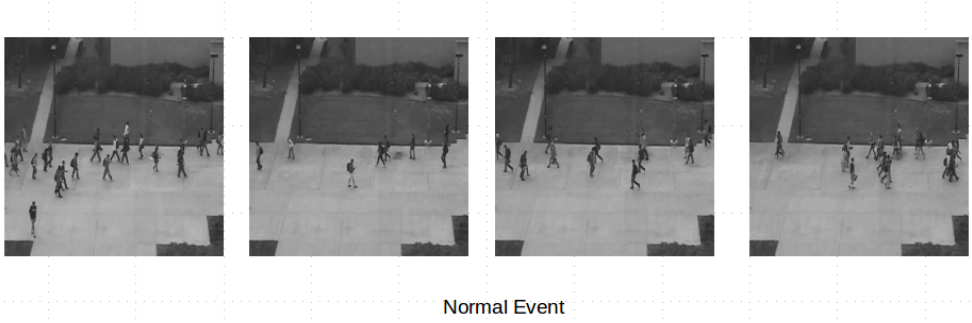
FIGURE 3.6: Motion classification

3.1.5 Experiments results

UCSD Peds2 and UMN Patil and Biswas, 2016, figure 3.10 are two different anomaly detection datasets. UCSD Peds2 consists of a video footages of crowded pedestrian walkway. It contains both normal and abnormal events like walking movement of bikers, skaters, cyclist and small carts. In the walkways, the motion of the pedestrians in an unexpected area is also considered as an anomalous event. It contains 16 training and 12 testing video samples and provides frame-level ground truth which helps us to evaluate the detection performance with comparing our method with others stat-of-the-art anomaly detection methods. In the other hand, The UMN dataset is consisted of 3 scenes: lawn (1450 frames), indoor (4415 frames) and plaza (2145 frames). It has two events: people walking which is considered as normal event and people running which is considered as abnormal event. The ground truth is provided in the video frames that need to be extracted to evaluate the performance.

Initialization:
N=number of Frames
$Train_1 = [\,]$
$Train_2 = [\,]$
for *i=1:N* **do**
  # F_i: The ith Frame
  $D1 = Feature\_extraction\_shapes\_FC1(F_i)$
  $Train_1 = [Train_1; D1]$
  $\#size(Train_1) = N$ vectors of 4096 values
  $D2 = Feature\_extraction\_shapes\_FC2(F_i)$
  $Train_2 = [Train_2; D2]$
  $\#size(Train_2) = N$ vectors of 4096 values
  $HOF_{Train} = Feature\_extraction\_motion(F_{i-1}, F_i)$
  $\#size(HOF_{Train}) = N$ vectors of 2 values
end
$Model1 = Train\_OC\_SVM(Train_1)$
$Model2 = Train\_OC\_SVM(Train_2)$
Anomaly detection:
for *Each new Frame* F'_i **do**
  $D1_{Test} = Feature\_extraction\_shapes\_FC1(F'_i)$
  $D2_{Test} = Feature\_extraction\_shapes\_FC2(F'_i)$
  $HOF_{Test} = Feature\_extraction\_motion(F'_{i-1}, F'_i)$
  #Abnormal shapes detection
  $Label1 = Predict(Model1, D1_{Test})$
  $Label2 = Predict(Model2, D2_{Test})$
  #Abnormal motion detection
  if $HOF_{Test}[2] > Threshold$ **then**
    $Label3$=*Abnormal*
  else
    $Label3$=*Normal*
  end
end
if *(*$Label1 = Abnormal$ *Or* $Label2 = Abnormal$ *Or* $Label3 = Abnormal$*)* **then**
  $Label = Abnormal$
else
  $Label = Normal$
end
$Label = Post\_processing(Label)$

Algorithm 1: Algorithm of Anomaly Detection



Normal Event

Abnormal Event

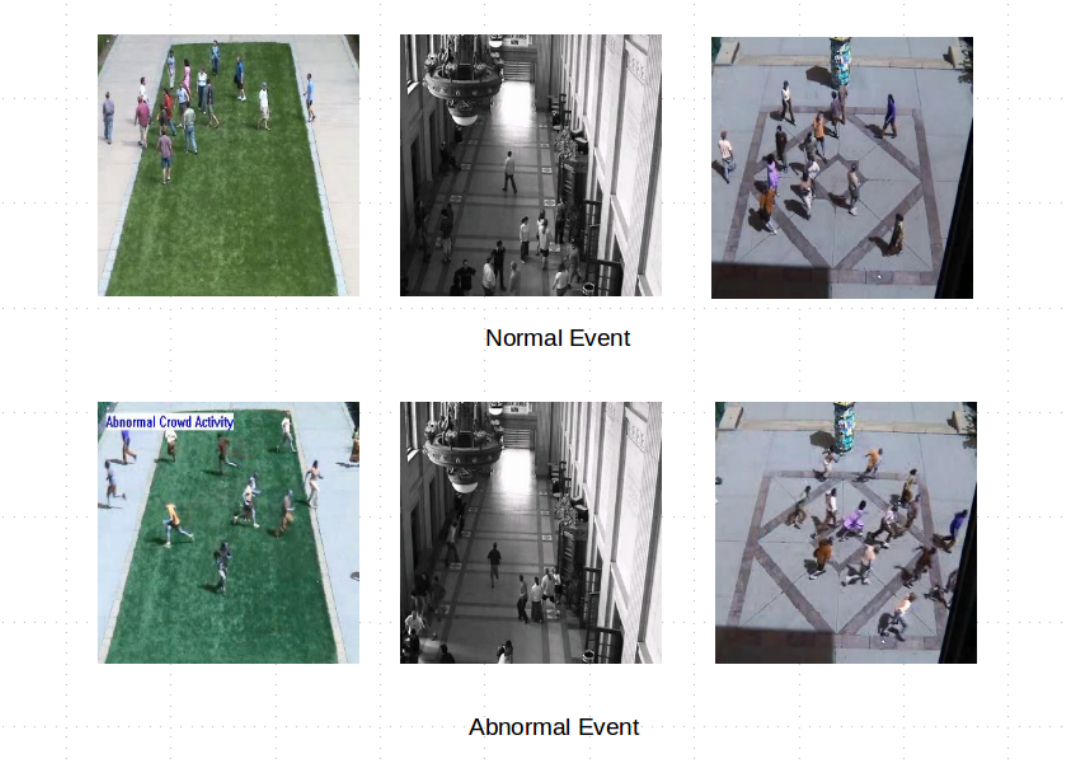


FIGURE 3.8: UMN Dataset

In order to evaluate the performance of the proposed method, we used the measure of ERR (Error), calculated with the following equation :

$$ERR = \frac{FP + NF}{FN} \tag{3.13}$$

- FP : False Positive, representing the false alarm.
- FN : False Negative, representing the miss detection.
- NF : Number of frames in each folder.

Our results in each folder of USCD Ped 2 is presented in the following table :

It could also be noted for each test folder that we have used the images of corresponding train folder. Despite we have trained the SVM with a small dataset around 150 frames at each folder, our method achieved good results comparing with others method summarized in TABLE III. It is reached a high performance and proved its efficiency comparing with the most of the state-of-the-art methods. The one-class SVM classifier is robust to the training dataset size, which allows its applicability for very large video datasets. Moreover, this hybrid system decreases the false negatives (miss detections) by dissociating of all descriptors decisions.

3.1.6 Post-processing

Moreover, we propose a post-processing to decrease the erroneous decisions taken by each classifier. So, we define a new decision function D_{in} based on the n previous decisions taken

TABLE 3.1: Results in USCD Ped 2 dataset

Folder	nbr frames train	nbr frames test	FP	FN	ERR
1	120	180	12	0	6.6%
2	150	180	95	0	52.7%
3	150	150	3	0	2%
4	180	180	20	0	11.11%
5	180	150	20	0	13.3%
6	150	180	20	4	13.3%
7	150	180	45	0	25%
8	120	180	0	0	0%
9	180	120	0	0	0%
10	180	150	0	2	1.3%
11	180	180	0	0	0%
12	180	180	88	0	48.8%

classification noted T_i and based on hypothesis that an anomaly can not appear or disappear abruptly (Fig. 4). The following TABLE I summarizes all the possibilities for n=3.

TABLE 3.2: Table of truth

T_{i-2}	T_{i-1}	T_i	D_{i3}
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

- 0 : Frame contains anomaly
- 1 : Frame does not contain anomaly

The post-processing has proven its efficiency by decreasing the error by 1.2% for UCSD Peds2 dataset and 1.7% for UMN dataset. It enhances the stability of our classify to detect the anomalies (Fig. 5). To evaluate the using of two fully connected layers the TABLE IV summarizes the results on UCSD Ped2 dataset using only one fully connected layer at each time. We note that using two fully connected layers has more efficiency and enhance our results by decreasing the total error by 3.45% at average. However, it should be noted that we can not use the third fully connected layer because it represents the probability membership vector for each 1000 classes.

Our results in scene of UMN is presented in the following table :

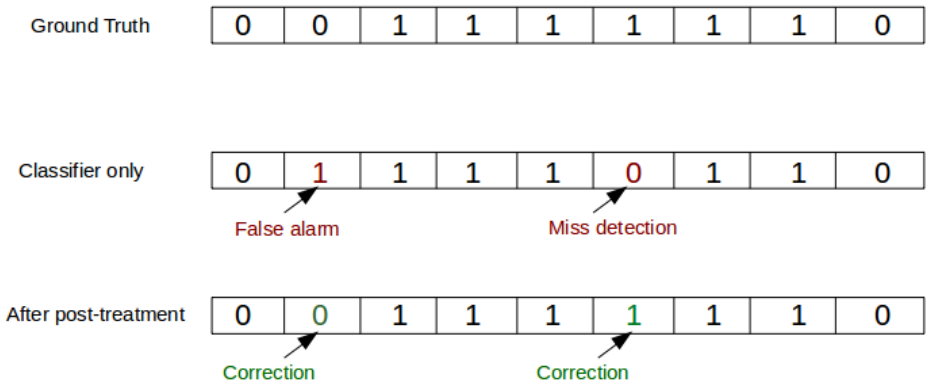


FIGURE 3.9: Example of post-processing

TABLE 3.3: FC1 and FC2 performance on USCD Peds2

Method	ERR
FC1+HOF	16.3%
FC2+HOF	20.19%

TABLE 3.4: Results in UMN dataset

Scene	ERR
Lawn	3.25%
Indoor	3.25%
Plaza	4%

These both tables (TABLE V and the TABLE VI) summarize the results on UMN dataset and show high performance of our method in anomaly detection compared with state-of-the-art methods.

3.2 Fine tuning CAE for deep One class

3.2.1 Introduction

Transfer Learning in the context of neural networks has allowed us to develop an effective methods for detecting abnormal events, taking into consideration the challenges of intelligent video surveillance systems. In particular, we have demonstrated that a pre-trained neural networks on large databases for action classification are particularly adapted for the characterization of shapes and movements in video footage, which makes them effective tools for the detection of abnormal events. However, the approach's reliance on networks pre-trained in a supervised mode on large, labeled databases can be detrimental. Indeed, the fact that the network used is developed and trained for another classification task may induce a mismatch with the target data. As an example we can cite the color characteristics contained in the network representations that lead to the classification of images and not used in our applications for detecting abnormal events. We can also suppose that the size of the field receptors of each

network layer is not adapted to the size of the objects present in the images of the monitored scene. On the other hand, the use of pre-trained networks imposes a certain inflexibility on our approach and considerably reduces its prospects for improvement. Unsupervised learning could be an alternative to transfer learning and could remove the dependency of our approach on large, labeled databases. Unsupervised learning is a sub-domain of machine learning which, as its name implies, involves learning characteristics from unlabeled data. Unsupervised learning may be particularly suitable for detecting abnormal video events, because this task is characterized by the unavailability of abnormal data during the training phase. In this section, we will explore a strategy to exploit convolutional neural networks in unsupervised mode for the extraction of spatio-temporal representations, usable for anomaly detection in video footage.

3.2.2 Auto-encoders

In unsupervised neural networks, the AutoEncoder (AE) ,Rumelhart, Hinton, and Williams, 1986,Bengio, 2012, is probably one of the most popular neural networks. The auto-encoder is a network of fully connected neural with one or more hidden layers. figure 3.X.

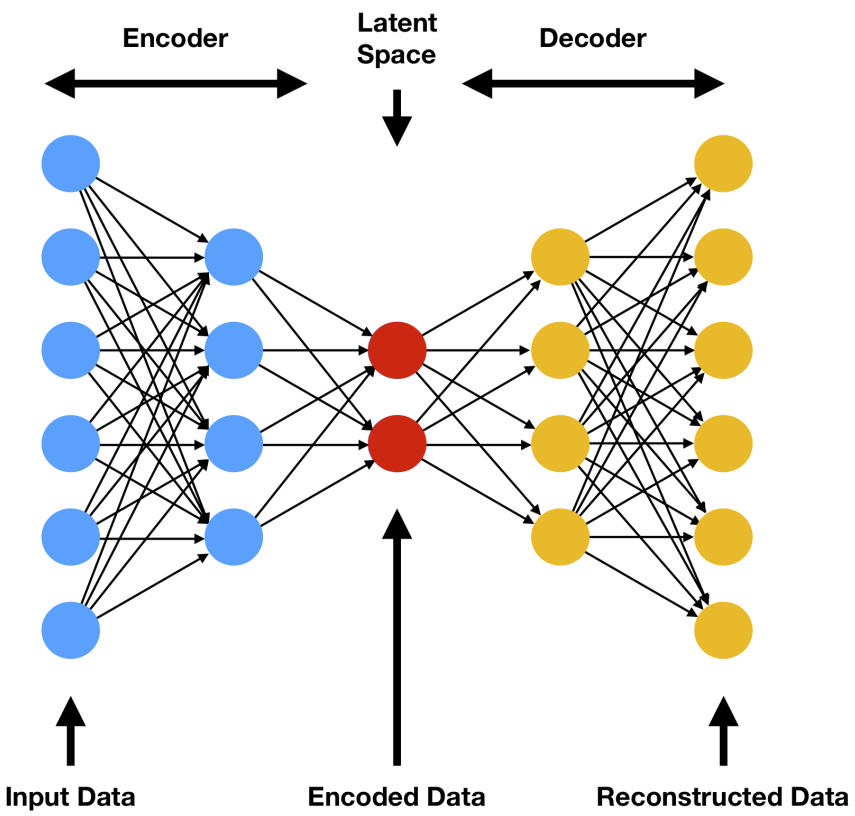


FIGURE 3.10: Autoencoder ; all layers are fully connected layers and latent Space is the bottleneck layer

It is trained to reconstruct an input data through an intermediate representation, often of reduced size. The objective is to extract characteristic descriptions in order to learn a more compact representation of the input data. For an input data $X \in R^d$ the AE extracts through the encoder a latent representation Z using the equation :

$$B = g(WX + b) \tag{3.14}$$

Where, B the reduced data encoded in the Bottleneck layer. W and b are the weights of the encoder part. The AE then reconstructs X , using the decoder from the bottleneck representation with the equation :

$$X' = g(W'B + b') \tag{3.15}$$

In general, the weights of the W' are defined as the transposed weights of the W ; $W' = W^T$. In this way the AE associates to each input data X a compact representation at the bottleneck layer B and a reconstruction X'. AE training is generally done by minimizing the reconstruction error, in particular through a loss function such as the mean squared error (MSE) given here below :

$$E(X,X') = \frac{1}{n}\sum(x_i - x'_i)^2 \tag{3.16}$$

3.2.3 Convolution Auto-encoders

The Convolutional AutoEncoder (CAE) [96] is an unsupervised neural network, similar to the traditional autoencoder, it also consists of an encoder and a decoder. Figure 3.7.

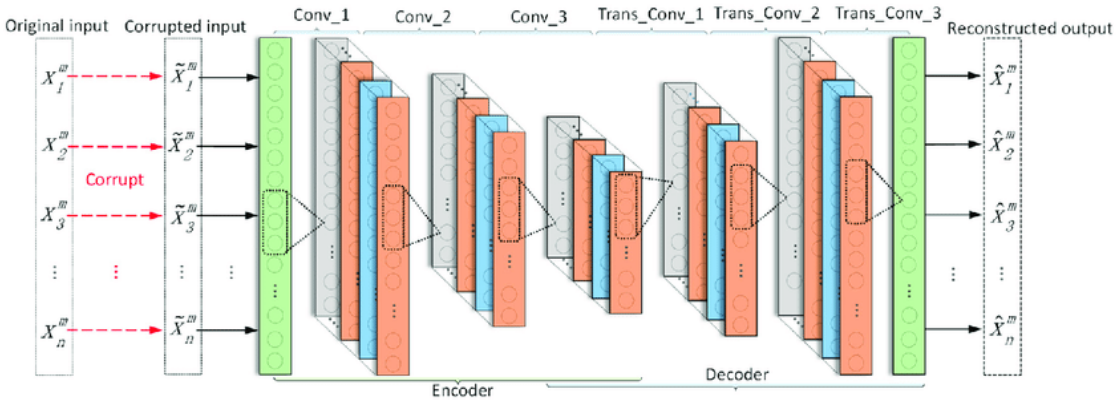


FIGURE 3.11: Structure of a 1-D denoising convolutional auto-encoderLiu et al., 2019.

The encoder extracts representations from the input data in a hierarchical structure through successive convolution layers. These representations are then used by the decoder to reconstruct the input data through multiple deconvolution layers. The traditional AE is based on its fully connected architecture, which limits its ability to represent 2D structures with important spatial relationships, such as images. Indeed, as we have previously motioned, fully connected layers overlook the spatial relationships in the input data. The convolutional auto-encoder, on the other hand, is based on local connections with shared weights in the same way as the CNN. This aspect allows to extract localized patterns in different regions of the input image. These features make CAE a tool particularly adapted to image processing tasks such as anomaly detection since anomalies occur most often in a localized pattern in the monitored scene. In a CAE with only one convolution and deconvolution layer, for each input mono channel image X. The bottleneck representation B of the k^{th} feature map obtained through the convolution layer is given by the equation :

$$B_k = f(X_k * X + b_k)) \tag{3.17}$$

Where, f is the activation function, X_k and b_k are the weights of the CAE related to k^{th} convolution filter. Than the reconstructed output X' is obtained as follow :

$$X' = \sum_{p=0}^{k} f(W'_p * B_p + b'_p) \tag{3.18}$$

Where, where K is the number of feature maps in the bottleneck representation, W'_p and b'_p are respectively the weights and the bias of the k^{th} deconvolution filter.

3.2.4 Mahalanobis distance classifier

The Mahalanobis distance is a measure of distance introduced by Prasanta Chandra Mahalanobis in 1936. it allows to determine the similarity between a known data set and new observations. Given a set of data X= (x_i, i = 1..N), $x_i \in R^d$. the Mahalanobis distance between y and set X is given by the equation

$$Mahalanobis = \sqrt{(y-\mu)S^{-1}(y-\mu)^T} \tag{3.19}$$

where, $\mu \in R^d$ and $S \in R^{d\times d}$, respectively the mean and the covariance matrix of the set X. As compared to the Euclidean distance, the Mahalanobis distance takes into account the variance and the correlation of the data set. The inverse of the covariance matrix gives less weight to the most dispersed components (largest variance) while the Euclidean distance treats all the components independently and in the same way. Because of its ability to measure the similarity of new observations to a known set, the Mahalanobis distance has often been exploited for the detection of outliers.

3.2.5 Deep One Class with fixed point

In a task of classification or pattern recognition in general, the representations must satisfy two essential criteria Perera and Patel, 2019. The first criterion concerns the inter-class distance, this distance must be large enough to be able to dissociate the elements belonging to these classes. This criteria is generally ensured by the representativeness (descriptiveness) of the representations. The more the representations are able to describe the data precisely, more the representations of learning elements in different classes are distant. The second criteria concerns the distance intra-class, the latter is assimilated to the compactness of the representations belonging to the same class. The ideal case would be to obtain the same representation for different data belonging to the same class, this would greatly simplify their isolation from the other classes. Nevertheless, in real cases, one is satisfied to obtain a compact enough cluster to represent the samples of the same class. By using convolutional auto-encoders, we have taken into consideration the criteria of representativeness, but have omitted the second criterion relating to compactness. Indeed, training a network to reconstruct input data through feature extraction ensures those features are sufficiently representative to describe the data. we therefore propose in this section to study different strategies in order to respect both the representativeness and the compactness of the extracted characteristics. We will then introduce an original learning method that ensures both the representativeness and the compactness of the representations. In the next chapter we will also introduce an end-to-end new method for deep one class anomaly detection.

Deep One class

By training a multi-class neural network, the two criteria compactness and representativeness are satisfied. Indeed, during the learning process, the network will progressively merge the representations of the same class while separating the representations of the different classes in the features space in order to be able to dissociate them. This will eventually lead to

the creation of distinct and easily separable clusters for the various classes. On the other hand in a one-class problem, such as detection of abnormal video events. the task is more difficult, given the fact that only one class of data is available, the network may eventually learn a trivial solution because there is no penalty for misclassification. In this case, the learned representations will be compact but not descriptive. which can lead to confusion between the samples of the target class and the outliers. Recent work in the literature has addressed this problem. In Ruff et al., 2018, a method called Deep SVDD is introduced for one-class learning. The method is inspired by the SVDD (Support Vector Data Description), it consists in training a neural network, with the aim of extracting representations grouped in a hypersphere of minimal volume. The learning of the network is done through the following objective function :

$$\min_{W} \frac{1}{N} \sum_{i=1}^{n} \left\| \varphi(x_i, W) - c^2 \right\| + \frac{\lambda}{N} \sum_{l=1}^{L} \|W\|_F^2 \tag{3.20}$$

Where, $\varphi(x_i, W)$ is the representation extracted by the network for a given dataset x_i. The second term λ is a hyperparameter regulator. Through this objective function the neural network, during the learning process, will contract the sphere minimizing the mean distance of all the data representations from the center. Thanks to this learning process, the network will learn parameters allowing to link each data of the target class to a representation included in the center hypersphere. However, the criteria of representativeness is not taken into consideration. In the next section we introduce new method of deep one class taken in account both representativeness and compactness into consideration. Indeed, the network can learn a trivial solution that will generate the extraction of representations belonging to the hypersphere even if these representations are relative to abnormal data.

A strategy to consider both simultaneously. criteria was proposed in Perera and Patel, 2019. The authors propose an architecture with two parallel networks continuously sharing the same weights. The two networks are trained with two loss functions, a compactness loss and a descriptiveness loss. The learning of the two networks involves two distinct databases, a first target database (target dataset) containing a single class (the target class) and a second reference database (reference dataset) containing several classes of images. During training, a batch of images extracted from the target database is introduced into the first network, which generates a loss of compactness. Simultaneously a second batch, extracted from the reference database is introduced into the second network, which generates a representativeness loss, The two losses are then added together and used to update the weights of the two networks identically, the process is thus repeated until the two networks converge. This architecture allows the criteria of descriptiveness and compactness to be satisfied at the same time. Nevertheless it remains strongly dependent on the reference database. In the next section we introduce our method of deep one class without using any reference database. Inspired by these two works, we propose a one-class learning method, using only a target database. Indeed, thanks to this method, the two criteria of compactness and representativeness will be respected without involving other related databases. This method provides deep representations that are both compact and representative. To achieve this, we propose to use also two different loss functions. In comparison to Perera and Patel, 2019, the two functions of losses are only calculated using data from the target database. We propose to integrate in a convolutional auto-encoder two loss functions, a reconstruction loss and a compactness loss. other works have been proposed for neural networks one class classification however these networks use a second external database, which creates a dependency on this external database. whereas our method is based only on the target class data.

3.2.6 proposed Method

Architecture

One-class classification is a machine learning problem that has received important attention by many researchers in different fields such as novelty detection, anomaly detection, and medical imaging. Nevertheless, the lack of data in the training phase reduces the depth of network architecture which in turn reduces the representativeness of features. To solve this weakness we propose to fine-tuning a pre-trained CAE for a one-class training objective constructed from VGG 16 CNN which is achieved 92.7% top-5 test accuracy. The database used to train VGG 16 CNN is ImageNet which is a dataset of over 14 million high-resolution images belonging to 1000 classes. The images were collected from the web and labeled by humans using Amazon's Mechanical Turk crowd-sourcing tool. We freeze the first layers of convolutions to properly exploit the richness of the database with which the CNN was trained (Figure 3.8). The objective of the convolution operation is to extract the high-level features from the input image. Our architecture need not be limited to only one convolution layer. Conventionally, the first convolution layer is responsible for capturing the Low-Level features such as edges, color, gradient orientation, etc. With added layers, the architecture adapts to the High-Level features as well, giving us a network that has the wholesome understanding of images in the dataset, similar to how we would. So, we construct the encoder part of our CAE architecture based on convolutions layers of pre-trained CNN VGG16. We freeze the first convolutional block of VGG 16 and we keep the others convolutional blocks trainable (Figure 3.9). In the hand, the decoder part is a plane network made up of four 2D-deconvolution layers to be able to reconstruct the original frames, Its hyper-parameters is given in (Table 1).

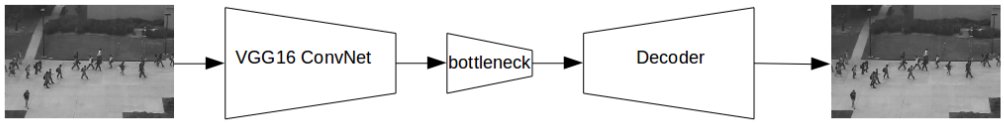


FIGURE 3.12: 2D-CAE based on pre-trained CNN VGG16 ConvNet

TABLE 3.5: Hyper parameter of added layers

Input size	Layer type	Filter Number	Kernel size	Strides	Activation	Output size
[7, 7]	2D-convolution	512	[3,3]	[2,2]	Relu	[3, 3]
[3, 3]	2D-deconvolution	256	[5,5]	[3,3]	Relu	[11,11]
[11,11]	2D-deconvolution	128	[5,5]	[2,2]	Relu	[35,35]
[35,35]	2D-deconvolution	96	[7,7]	[2,2]	Relu	[109,109]
[109, 109]	2D-deconvolution	1	[8,8]	[2,2]	linear	[224, 224]

Similar to the traditional auto-encoder, the CAE is composed of two parts. The encoder part which is a sequence of convolutional layers aims to extract compressed data of input image at the bottleneck layer and the decoder part which is successive of deconvolutional layers aims to reconstruct the input data from compressed data at bottleneck layer. The CAE can reconstruct better the data with was trained than the data that have ever seen, so the bottleneck layer must be reduced and representative as possible which in reality presents a compromise, many tests are done to select properly the bottleneck dimension (Table 3.1). A non-linear activation function is used at the convolutional and deconvolutional layers to obtain more useful and robust representations, except for the last deconvolution layer we used linear activation function due to the range of our input data which is [-255,255]. Our

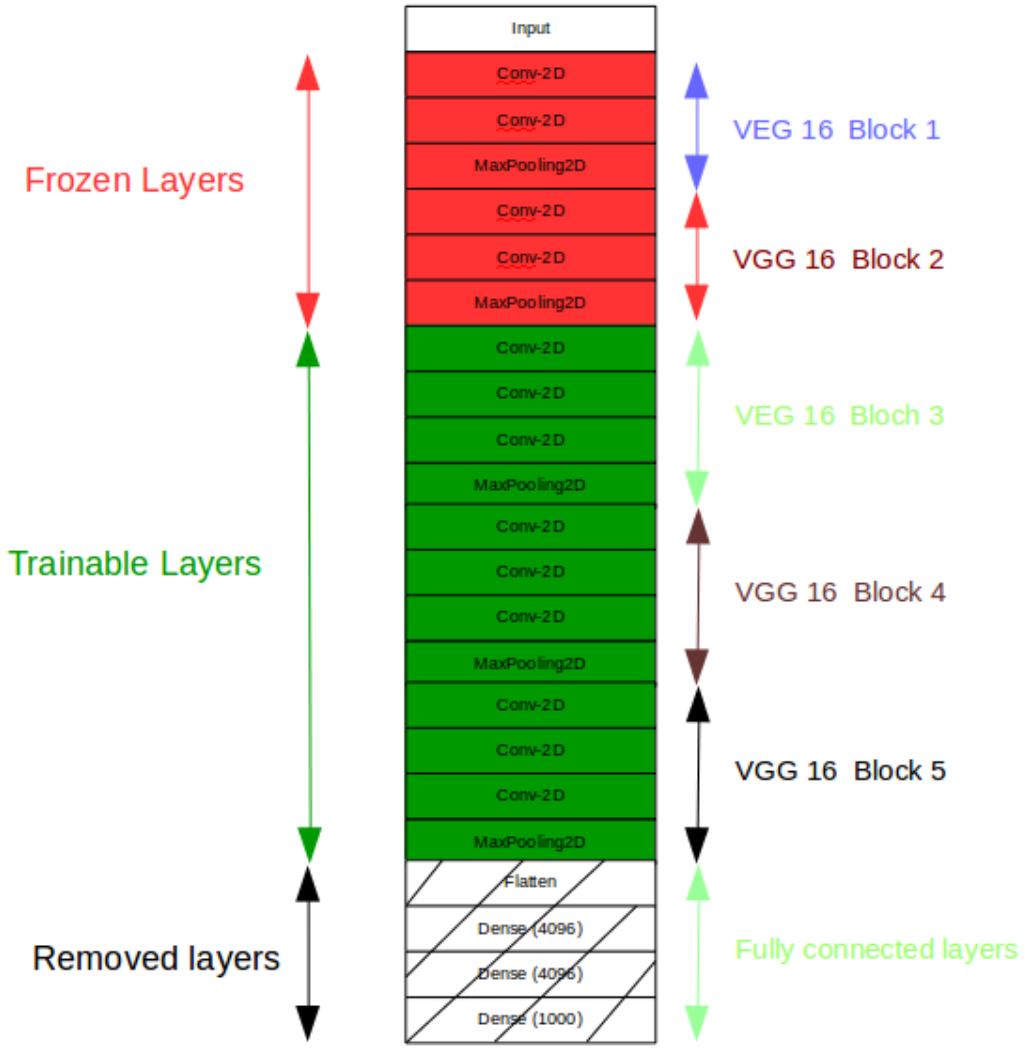


FIGURE 3.13: VGG 16 architecture used for fine-tuning one class objective

architecture consists of two parallel CAEs constructed as mentioned above. The first CAEs are trained on original images to be able to detect any abnormalities in shapes and the second CAEs are trained on optical flow representation aim to detect any abnormal motion relative to training (Figure 3.10).

Training

The training phase aims to obtain a model capable to get representative and compact features of normal images for easy classification. We can ensure that in two methods; the first method (Figure 3.11) is to do training in cascade objectives by training only at the beginning with the reconstruction objective and after a few epochs we extract a representative point denoted "c" of features of the dataset which with our model is training at bottleneck layer as the mean of features. Then, we do training only with the compactness objective and we fix the point c as the target of our new features. The disadvantage of this training method is that the representativeness of the images is not robust but it gives very compacted features. To remedy this flaw, a second training method is proposed with pseudo-parallel objectives (Figure 3.12), we start the training with only reconstruction objective then as we have done

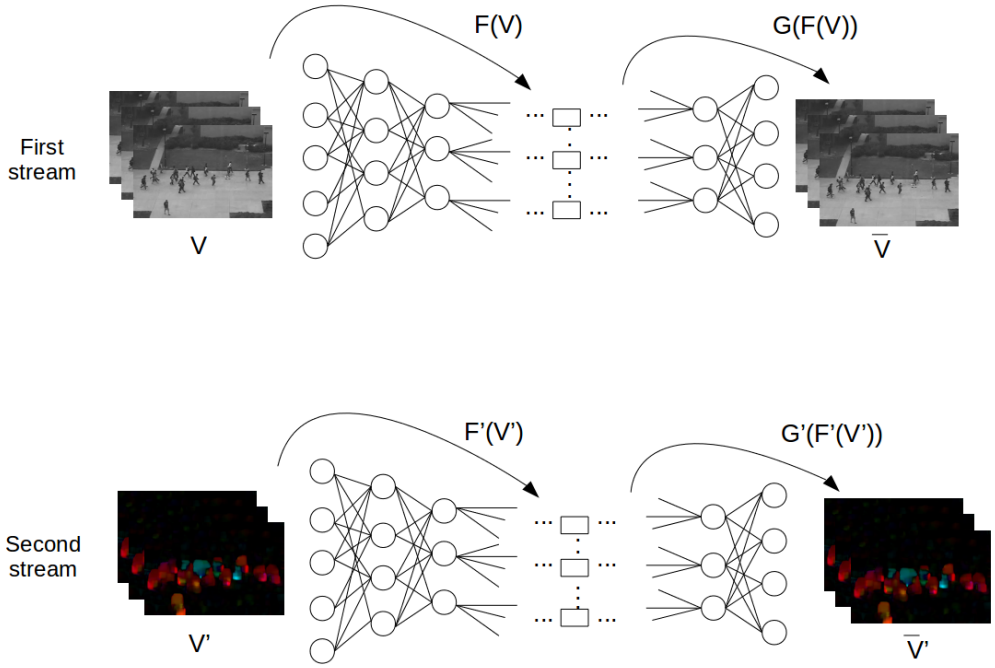


FIGURE 3.14: Two stream learning

at the first method we extract a fixed point "c" as the target of features then we continue the training with both compactness and reconstruction objectives to get robust model.

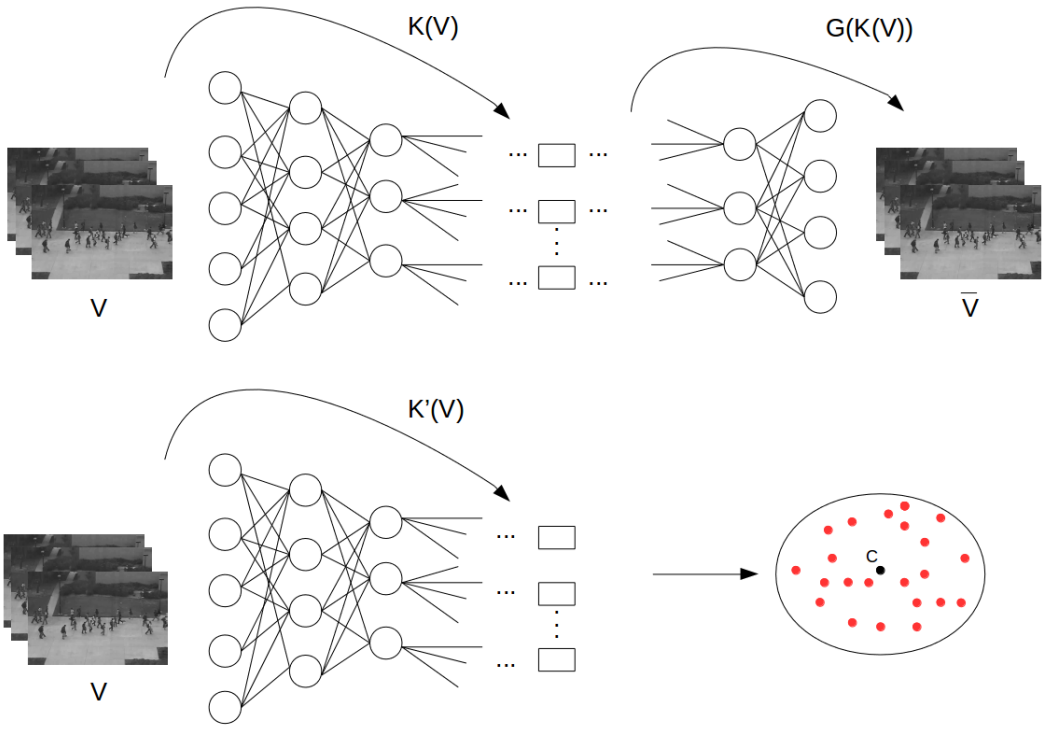


FIGURE 3.15: The first training method : Cascade objectives

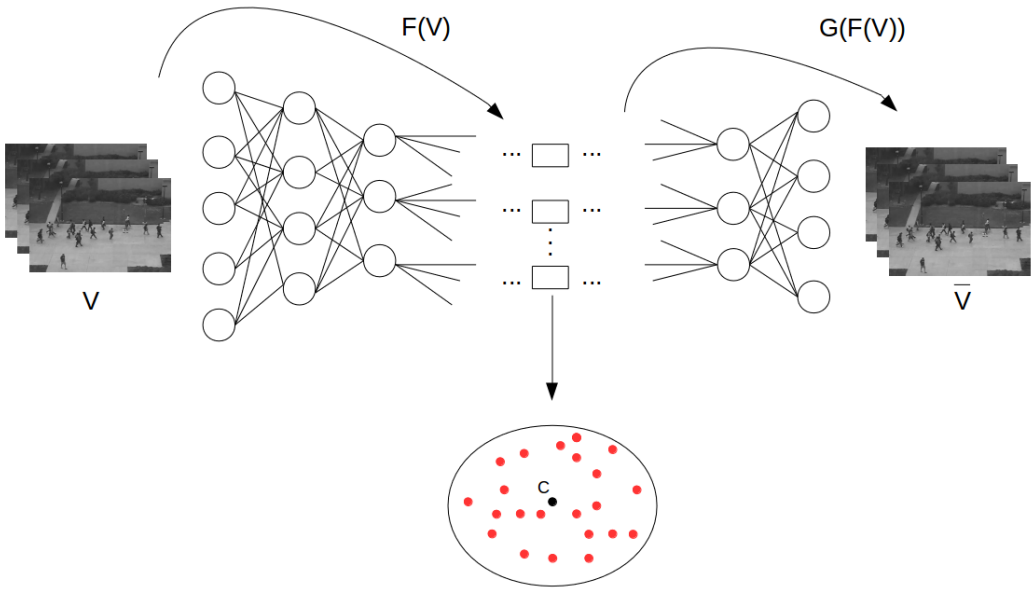


FIGURE 3.16: The second training method : pseudo-parallel objectives

During the training phase (Figure 3.12), both 2D-CAE are trained, one is trained with a stream of a sequence of original images and the other is trained with a stream of a sequence of optical flow representation. The optical flow is the pattern of apparent motion of image objects between two consecutive frames caused by the movements of the object. We have used a color code for better visualization. (Figure 3.13) shows some samples of images and optical flow images.

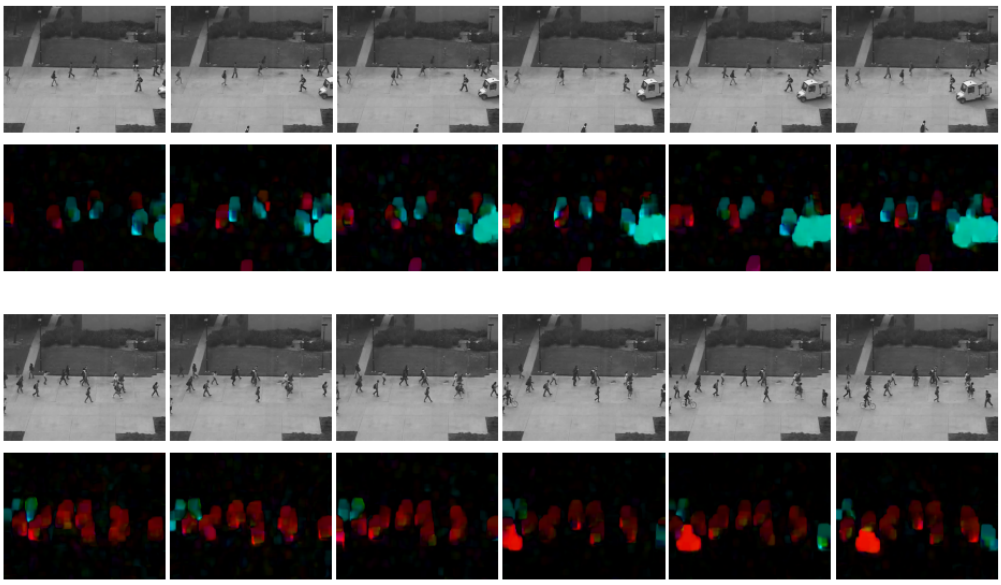


FIGURE 3.17: examples of optical flow representations and original images

Representativeness loss : $\mathbf{L}_r$

The aim of representativeness loss is to evaluate the capacity of the learned feature to generalize normal class. The representativeness loss increases the capacity of our model to raise the distance inter-classes.

$$L_r = \frac{1}{n}\sum_{i=1}^{n}(V - \hat{V}) \tag{3.21}$$

Compactness loss : $\mathbf{L}_c$

The objective of compactness loss is to tight all the features used during the training phase belonging to the normal class. Compactness loss evaluates the similarity between each feature vector and the fixed point 'C'. It is used to decrease the intra-class variance of the normal class.

$$L_c = \frac{1}{n}\sum_{i=1}^{n}(F(V) - M) \tag{3.22}$$

To perform back-propagation using this loss, it is necessary to assess the contribution each element of the input has on the final loss. For each ith sample F(V)= $\{Fv_{i1},Fv_{i2},...Fv_{ik}\}$ $\in R^k$ and the fixed point as m_i= $\{m_{i1},m_{i2},...m_{ik}\}$, we define the gradient l_c with respect to the input Fv_{ij} is given as,

$$\frac{\partial L_c}{\partial Fv_{ij}} = \frac{2}{(n-1)n_k}[n \times (Fv_{ij}) - \sum_{k=1}^{n}(Fv_{ik} - m_{ik})] \tag{3.23}$$

Testing

The proposed testing procedure aims to classify features of testing images as normal or abnormal based on the Mahalanobis distance threshold. Both motion and shapes features vectors noted respectively F(v)= $\{Fv_{i1},Fv_{i2},...Fv_{ik}\} \in R^k$and F'(v')= $\{F'v'_{i1},F'v'_{i2},...F'v'_{ik}\} \in R^k$ are extracted from trained encoders parts to be concatenated into one vector. Then we apply PCA to this vector to reduce dimension and to extract important information noted X = PCA ([F(V) ; F'(V')] = $\{X_{i1},X_{i2},...X_{ik},X_{i1}\} \in R^p$ when p $<$ 2 $\times$ k (Figure 3.14). Using PCA is made the calculation of the covariance matrix Q faster and not complicate.

For each new feature vector X_{test} we calculate a Mahalanobis distance between each feature vector and $\bar{X}$ as given :

$$d = (X_{test} - \bar{X}) \times Q^{-1} \times (X_{test} - \bar{X})^t \tag{3.24}$$

When $\bar{X}$ as the mean of X $\in R^p$ and Q $\in R^{p\times p}$ as its covariance.

The classification process is carried out according to the following process: In the first step, we extract feature vectors X =$\{x_i\}$,$x_i \in R^{512}$ from the normal training examples, the mean M and the inverse of the covariance matrix Q of X are then calculated. In the second step, we evaluate each feature vector x_j of the testing frames with Mahalanobis distance d_j using M and Q. This is represented in the following equation:

$$d\_j = (x\_j - M) \times Q \times (x\_j - M)' \tag{3.25}$$

The outlier vectors, which actually represents abnormal frames, are then picked by thresholding the distance. If the distance exceeds a threshold α, the vector x_j is considered as outlier

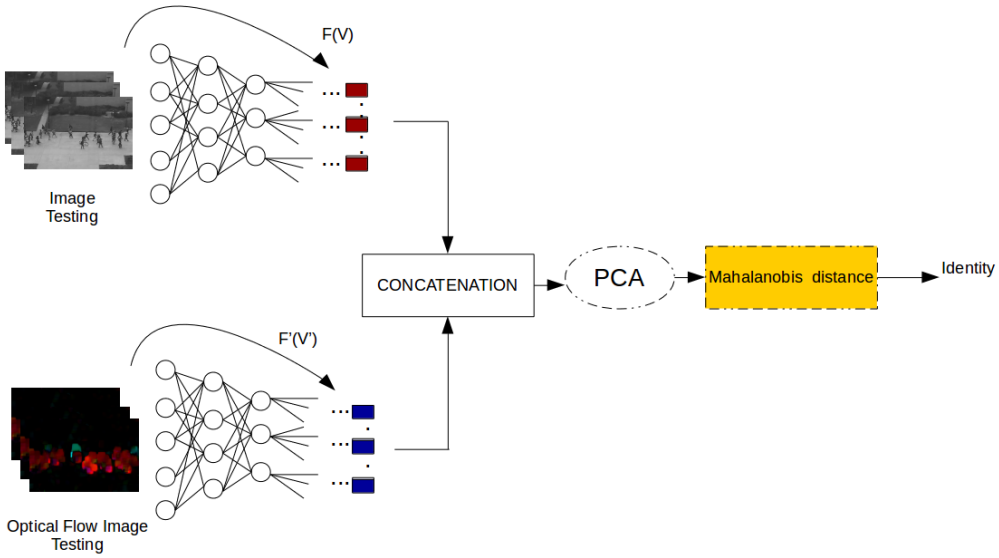


FIGURE 3.18: Classification flowcharts

and the frame p_j is labeled as abnormal, Eq (4.25).

$$p_j : \begin{cases} Normal & if \quad d_j \leq \alpha \\ Abnormal & if \quad d_j > \alpha \end{cases} \tag{3.26}$$

Experimental results

UCSD Peds2 and UMN are challenging anomaly detection datasets. Both of them contain normal events like people are walking and abnormal events like the walking movement of bikers, skaters, cyclists, and small carts in the case of Ped2, and people are running in the case of UMN. Ped2 contains 16 training and 12 testing video samples and provides frame-level ground truth to evaluate the detection performance by comparing our method with others stat-of-the-art anomaly detection methods. In the other hand, The UMN dataset has consisted of 3 scenes: lawn (1450 frames), indoor (4415 frames) and plaza (2145 frames) and the ground truth is provided in the video frames that need to be extracted to evaluate the performance.

We evaluate our different methods using (Error Equal Rate) EER and (Area Under Curve ROC) AUC as evaluation criteria. A smaller EER corresponds with better performance. As for the AUC, a bigger value corresponds with better performance.

Our two methods have the same results nearly, with a little advantage for the pseudo-parallel objectives method.

It proves the robustness to occlusion and high performance in anomaly detection compared with state-of-the-art methods. To visualize the important effect of the compactness loss function we extract from each feature extracted by our architecture two components by applying the PCA. These components are named later features for visualization. Figure 8 illustrates the results, just to better understand its effects, we will categorize our database into three classes.

- Normal images contains only normal events as mentioned in ground truth, this class represented by green points in figure 8.

- Confused images when a portion of anomaly start to appear and not a whole of the anomaly enter in the scene, this class is presented by blue points in figure 8.
- Abnormal images when a more of the half of anomaly enter in the scene, this class represented by red points in figure 8.

The Figure 3.15.1.a represents features for visualisations of our architecture trained with only representativeness loss, as we can see in this figures each of three classes reserved a region of space. Which is mean representativeness loss has increased the inter-classes distance between the three classes in an unsupervised way and using only the class of normal images (Class one). In order to decrease the intra-class distance for normal image we have used compactness loss. The figure Figure 3.15.1.b represents features for visualisations of our architecture trained with both representativeness loss and compactness loss. In this case, the normal images not only are reserved region in space but also are very tight and easy to separate from abnormal images.

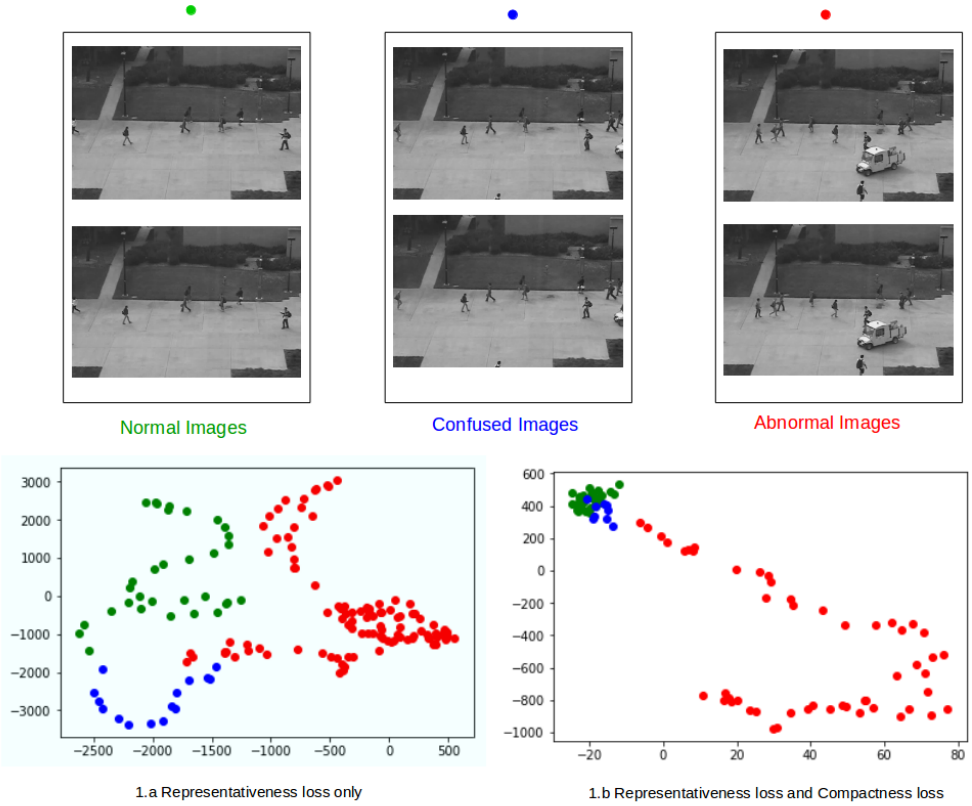


FIGURE 3.19: Compactness loss importance

Combining the two CAEs have decreased the EER from 17% to 11% which make the importance of using of optical flow image to represent the motion in each frames. The table 2 shows our results on Ped2 dataset and proves the robustness of our method compared to others state of the art methods.

Our results in scene of UMN is presented in the following table :

This table shows our results relatively at each scene. Despite that our model is trained on different scenes. It proves that our method have good efficiency for anomaly detection.

This table shows our results for UMN dataset, in this case we use one threshold for whole the dataset and its independent to the scenes. It proves that our method have good efficiency and robust for variation of scenes .

TABLE 3.6: EER comparison of UCSD Peds2

Method	EER
Mehran, Oyama, and Shah, 2009a	42.00%
Adam et al., 2008	42.00%
Kim and Grauman, 2009a	30.00%
Bertini, Del Bimbo, and Seidenari, 2012	30.00%
Zhou et al., 2016a	24.40%
Bouindour et al., 2017	24.20%
Mahadevan et al., 2010	24%
Hasan et al., 2016	21.7%
Reddy, Sanderson, and Lovell, 2011	20%
Sabokrou et al., 2015	19%
Li, Mahadevan, and Vasconcelos, 2013	18.50%
Ravanbakhsh et al., 2016	18%
Saligrama and Chen, 2012b	16%
Xu et al., 2017	17%
Sabokrou, Fathy, and Hoseini, 2016	15%
Boiman and Irani, 2007	13%
Roshtkhari and Levine, 2013	13%
Ravanbakhsh et al., 2017	14%
Chong and Tay, 2017c	12.00%
Sabokrou et al., 2018c	11%
Xiao, Zhang, and Zha, 2015	10.00%
Sabokrou et al., 2017	8.2%
Ours (Cascade)	**12%**
Ours(pseudo parallel)	**11%**

TABLE 3.7: Results in UMN dataset

Scene	EER	AUC
Lawn	3.17%	99.23%
Indoor	1.92%	99.37%
Plaza	1.11%	99.80%

TABLE 3.8: ERR comparison of UMN dataset

Method	EER
Mehran, Oyama, and Shah, 2009a	12.60%
Wu, Moore, and Shah, 2010.	5.30%
Li, Mahadevan, and Vasconcelos, 2013	3.70%
Saligrama and Chen, 2012a	3.40%
Cong, Yuan, and Liu, 2011	2.80%
Ours	**2.28%**

3.3 Conclusion

In this chapter, we have proposed two methods for the detection of abnormal events in video. These methods are both based on pre-trained neural networks trained on large databases for semantically different tasks of anomaly detection in video footage. Through these methods we were able to evaluate the interest of transfer learning in the context of abnormal events detection. The first method, based on a 2D FCN and an OC-SVM, allowed us not only to confirm the interest of using pre-trained neural networks for the extraction of features exploitable for the characterization of video events, but also to show the relevance of the fully convolutional architecture FCN for the detection of anomalies within images. Despite the interest of this method, we found that the pre-trained aspect of the network does not allow it to extract sufficiently robust spatio-temporal descriptors for the characterization of video events. Given the observation made based on our first method, we have oriented our work towards the unsupervised networks. Thanks to the combination of a pre-trained 2D network and custom Fully Convolution Decoder. We were in a position to propose a second method characterized by an unsupervised learning. This method has demonstrated very good abilities to detect abnormal events through the different tests performed on the public UCSD Ped2 and UMN datasets.

Chapter 4

Optical flow based deep learning in UAV videos

4.1 Introduction

Optical Flow is a descriptor of the apparent scene movement. It allows the extraction important information on the spatial arrangement of objects and their evolution. The optical flux is generally obtained by observing the variation of pixels in the time domain between two adjacent images. It is a displacement vectors, showing the movement of the pixels from the first image to the second in the orthogonal axes of the image. Usually the optical flux is caused by the displacement of foreground objects, but it can also be caused by the displacement of the camera or the combination of both. In this chapter we treat the problem of anomaly detection by drone video.

Figure 4.1. presents two consecutive images captured under exactly the same brightness conditions and at an interval of time dt. Considering $I(x,y,dt)$ a pixel at the first image and (dx,dy), the displacement of the pixel between the first and the second pixel images. Since this pixel is the same on both images and its intensity does not change, we can confirm the following :

$$I(x,y,t) = I(x+dx,y+dy,t+dt) \tag{4.1}$$

Assuming that the movement is small and using Taylor's series development we get :

$$I(x+dx,y+dy,t+dt) = I(x,y,t) + \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt + H \tag{4.2}$$

When H represents higher order terms.

Using equations (4.1) (4.2) :

$$\frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt = 0 \tag{4.3}$$

And by dividing by dt, we get :

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}y + \frac{\partial I}{\partial t} = 0 \tag{4.4}$$

$u = dx/dt, v = dy/dt$ are the components of the velocity or optical flow of $I(x,y,t)$
$frac\partial I\partial y, frac\partial I\partial x, frac\partial I\partial t$are the partial derivatives of the image at (x,y,t).

u and v are unknown variables and the equation 4.5 cannot be directly resolved. Several methods have therefore been proposed to solve this problem and one of them is the Farnebäck method.

4.2 Two-Streams FCN optical flow generating

4.2.1 Farnebäck Method

:

The Farneback method Farnebäck, 2003 is a method for optical flow estimation. The method assumes that the neighborhood of a pixel can be estimated using a quadratic polynomial, which gives the local signal model represented in a local coordinate system.

$$f(x) \sim x^T Ax + b^T x + c \tag{4.5}$$

Where A is a symetric matrix, b is a vector and c is a scaler. In a case of an optimal translation, the displacement d can be obtained by calculating the neighborhood polynomials on two consecutive images. The signal f_1 relative to the first image is given by the expression :

$$f_1(x) \sim x^T A_1 x + b_1^T x + c_1 \tag{4.6}$$

and the signal f_2, obtained after a global displacement d, is given by the expression :

$$f_2(x) = f_1(x-d) = (x-d)^T A_1 (x-d) + b_1^T (x-d) + c_1 \tag{4.7}$$

$$= x^T A_2 x + b_2^T x + c_2 \tag{4.8}$$

Assuming that the brightness is constant between the two images, we can define an equivalence between the coefficients of the two polynomials :

$$A_1 = A_2 \tag{4.9}$$

$$b_2 = b_1 - 2A_1 d \tag{4.10}$$

$$c_2 = d^T A_1 d - b_1^T d + c1 \tag{4.11}$$

assuming the matrix A is reversible, we can calculate the value of the displacement (i.e. the optical flow) as follows :

$$d = -\frac{1}{2} A_1^{-1} (b_2 - b_1) \tag{4.12}$$

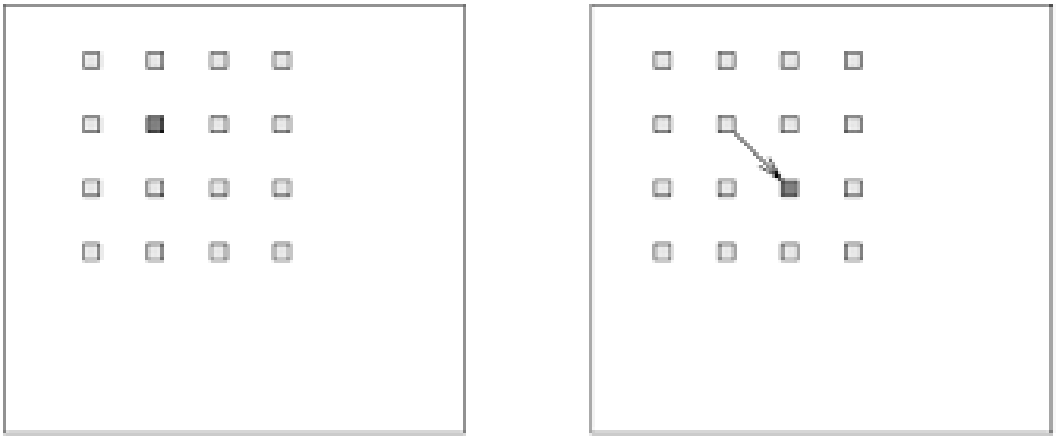


FIGURE 4.1: Pixel motion through two consecutive images

It is stipulated in Farnebäck, 2003, that the hypothesis that an integer signal is a single polynomial and the hypothesis of the global translation linking the two signals are quite unrealistic. The author of [16] thus considers local polynomial approximations and introduces the following approximations :

$$A(x) = \frac{A_1(x) + A_2(x)}{2} \tag{4.13}$$

$$\Delta b(x) = \frac{1}{2}(b_1(x) - b_2(x)) \tag{4.14}$$

to obtain the primary constraint :

$$A(x)d(x) = \Delta b(x) \tag{4.15}$$

$d(x)$ indicates the global displacement by a spatial variable displacement space. In principle, equation 3.31 can be solved on a punctual approach. However, to avoid noise, the author proposes to induce the hypothesis that the displacement spaces varies only slowly, so that information about a neighborhood of each pixel can be integrated. Thus the problem is to find dx that satisfies equation 4.14 as well as possible on a neighborhood I of x, which is to minimize the expression

$$\sum_{\Delta x \in I} = w(\Delta x) \left\| A(x + \Delta x)dx - \Delta b(x + \Delta x) \right\|^2 \tag{4.16}$$

Where $w(\Delta x)$ is a weight function for points in the neighborhood. The minimum is obtained for :

$$d(x) = (\sum wA^T A)^{-1} \sum wA^T \Delta b \tag{4.17}$$

4.2.2 Gradient descent

Gradient descent LeCun et al., 1998 is an iterative optimization algorithm widely used for training neural networks. It allows to find an optimum value for the network parameters and to minimize a differentiable loss function. The algorithm 1 is subdivided into two steps: forward propagation to compute the output vector and backward propagation where the partial derivatives of the loss function are computed with respect to the network parameters. Algorithm 1 formalizes the steps of the gradient descent for learning of the neural network parameters.

Let's consider a training database B containing N couples $(X;Y)$ where X is the input data and Y the label of the data, $J(\theta)$ a loss function to be minimized as represented by the network parameters θ, $\nabla J(\theta)$ the gradient of the function and the ηlearning rate. The algorithm consists to find the gradient of the error of each pair and to average it. Since the gradient is a vector pointed in the direction of highest increase in the error function, so shifting the parameters to the opposite direction of the gradient decreases the error. The learning rate allows to control this movement. In other hand, the gradient allows to define the gradients in the direction of the correction and the learning step allows to control the this correction. The choice of the learning rate is often crucial for network training, since a very high value will induce too great a change in weight, causing the optimal values to be missed. On the contrary a too small value will cause the slowdown of learning. In what follows we

will present an example of application of the gradient descent algorithm for the learning of a fully connected neural network.

Random initialisation :
$\theta = (\theta_0, \theta_1)$
while *Condition of stop not reached* **do**
  for *each couple (X,Y)* **do**
    S_k = Forward propagation of X_k
    $\nabla J_k(\theta)$ = Back propagation of X_k
  end
  $\nabla J(\theta) = \frac{1}{N} \sum_{k=1}^{n} \nabla J_k(\theta)$
  $\theta = \theta - \eta \times \nabla J(\theta)$
end

Algorithm 2: Gradient descent

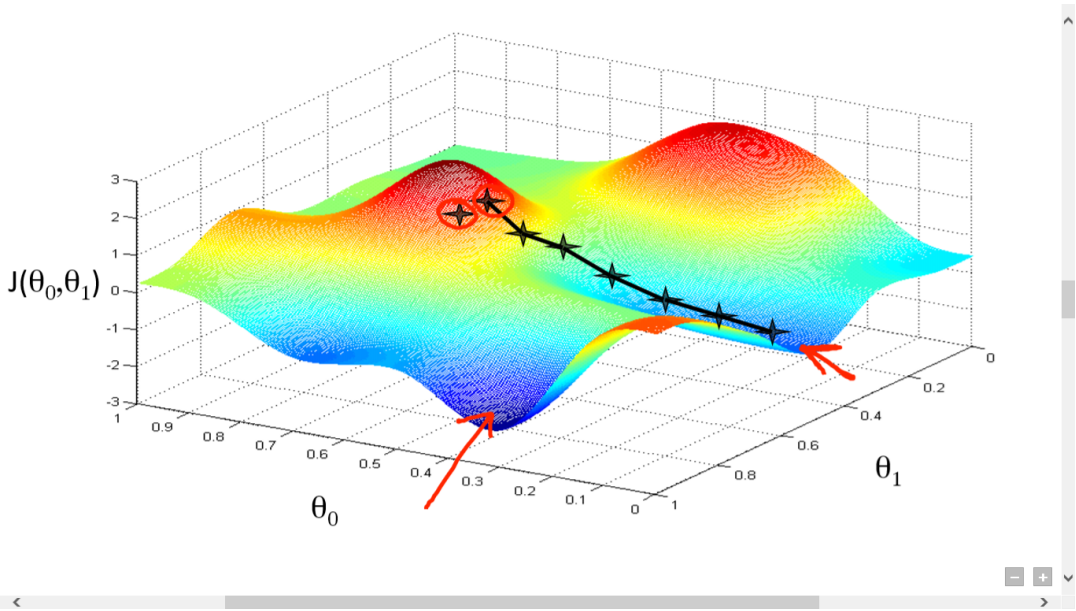


FIGURE 4.2: Gradient Descent

The learning of the network parameters using the gradient descent algorithm is done as follows :

- The first step is to initialize the network parameters. Usually random weights of a Gaussian distribution are used Krizhevsky, Sutskever, and Hinton, 2017.
- After the initialization of the weights, the first training data are propagated in the network in order to obtain an output. The final output of the network is obtained by hierarchically compute the activation of neurons through the different layers of the network (from the shallowest to the deepest layer). In the hidden layers the activation value a_j of a neuron j according to its inputs i is given by the following formula:

$$a_j = f(\sum_{i=1} w_{ji} a_i) \tag{4.18}$$

  Where w_{ji} is the weight of the connection neuron i with neuron j and f is the activation function.
  The same applies to the output layer, the activation of the S_k of a neuron k depending

on its inputs j is given by the equation :

$$S_k = f(\sum_{j=1} \eta_{jk} a_j) \tag{4.19}$$

- Once the output and the different network activations have been obtained, the next step is the calculation and back-propagation of the error.The error between the output of the network and the desired value Y is obtained using the following formula :

$$\delta_k = \frac{\partial E}{\partial S_k} = (y_k - s_k) s_k (1 - s_k) \tag{4.20}$$

The error is then backpropagated for the intermediate layers as follows :

$$\delta_k = \frac{\partial E}{\partial a_k} = a_k (1 - a_k) \sum_{k=1} \gamma_{ik} \delta_k \tag{4.21}$$

Once the errors are obtained, they are used in the last step which consists of updating the network weights. The output layer weights :

$$\gamma_{ik} = \gamma_{ik} - \eta \delta_k a_j \tag{4.22}$$

Weights of the intermediate layers :

$$w_{ij} = w_{ij} - \delta_j a_i \tag{4.23}$$

This learning process is repeated until the network converges.

4.2.3 Batch gradient descent

In the standard gradient descent method presented in Algorithm 2 the weights are only updated once the error gradient has been calculated for all training examples. Proceeding in this way is an efficient way to reach the minima at least noisy or random way possible. However, this strategy results in particularly slow learning and high computational complexity, especially for large learning sets. In order to overcome this, a different strategy called stochastic gradient descent SGD (Stochastic gradient descent) is used. In the SGD, for each iteration the gradient of the cost function is calculated and used to update the network weights for a randomly selected single training example. Since only one sample of the dataset is randomly selected for each iteration, the path taken by the algorithm to reach the minima is usually more noisy than for the standard gradient descent. However, the path taken by the algorithm is not of crucial importance, as long as optimal weight values are achieved with a significantly shorter learning time. A last strategy allows to find a compromise between the two previous strategies by proposing batch-based learning (mini-batch training). The training data are divided into batches and the network weights are updated at the end of each batch, Algorithm 3.

This strategy allows faster weight updates while limiting the amount of SGD-specific noise.

Random initialisation :
$\theta = (\theta_0, \theta_1)$
while *Condition of stop not reached* **do**
  for *each couple (X,Y) from the Batch B* **do**
    S_k = Forward propagation of X_k
    $\nabla J_k(\theta)$ = Back propagation of X_k
  end
  $\nabla J(\theta) = \frac{1}{N} \sum_{k=1}^{n} \nabla J_k(\theta)$
  $\theta = \theta - \eta \times \nabla J(\theta)$
end

Algorithm 3: Gradient descent by Batch

4.2.4 Experiments results

4.3 PROPOSED METHODS

Recently deeper two-streams convolutional networks have been applied successfully on action recognition. Based on this concept we propose a new efficient architecture composed of two FCNs to tackle the problem of anomaly detection in video into both different methods.

4.3.1 TS-FCN 1

Our proposed architecture consists of two parts: spatio-temporal FCN (ST-FCN) for learning representations from video frames, and optical flow FCN (OF-FCN) to strengthen the movement description of the learned representations. The learned two FCNs are obtained by training two convolutional auto-encoders (CAEs) in order to reconstruct video volumes and extracting the encoder part of each of them, Figure 4.3.

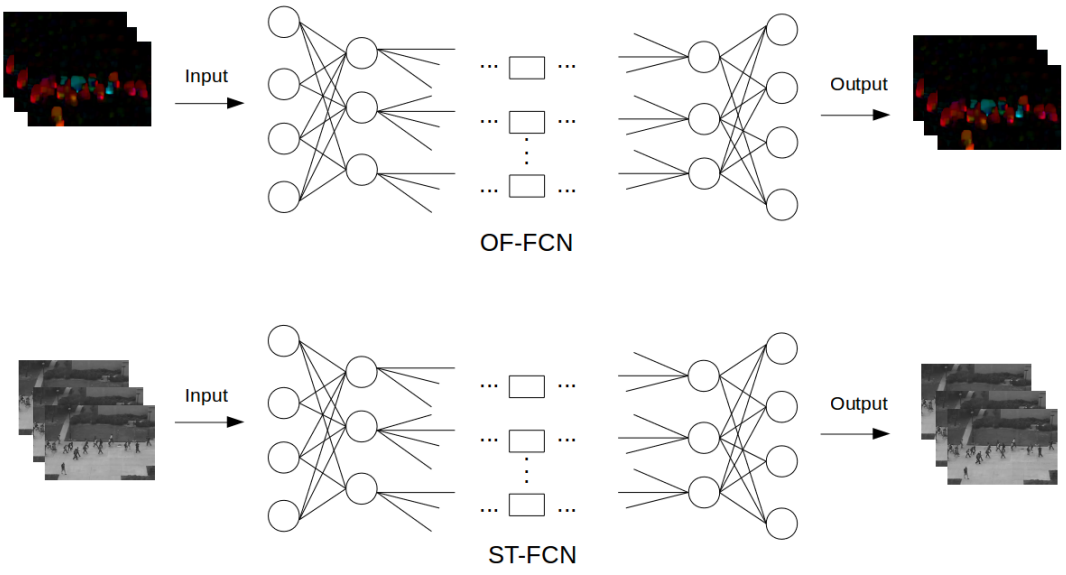


FIGURE 4.3: Our TS-FCN 1 Architecture

The spatio-temporal CAE and the optical flow CAE are respectively learned using normal training samples and corresponding optical-flow representations. Both CAEs have the same architecture and each of them is composed by four 3D convolution layers (encoder) and four 3D deconvolution layers (decoder) . The convolution layers encode representations from the input data while the deconvolution reflect the encoder part to reconstruct them. The spatio-temporal CAE takes as input 3D volumes of three consecutive frames F: $\{F_t; F_{t-1}; F_{t-2}\}$. The optical flow CAE, 3D volumes of three optical flow representations OF: $\{OF_t; OF_{t-1}; OF_{t-2}\}$, where OF_t is obtained by extracting the optical flow for each two consecutive frames Figure 4.4.

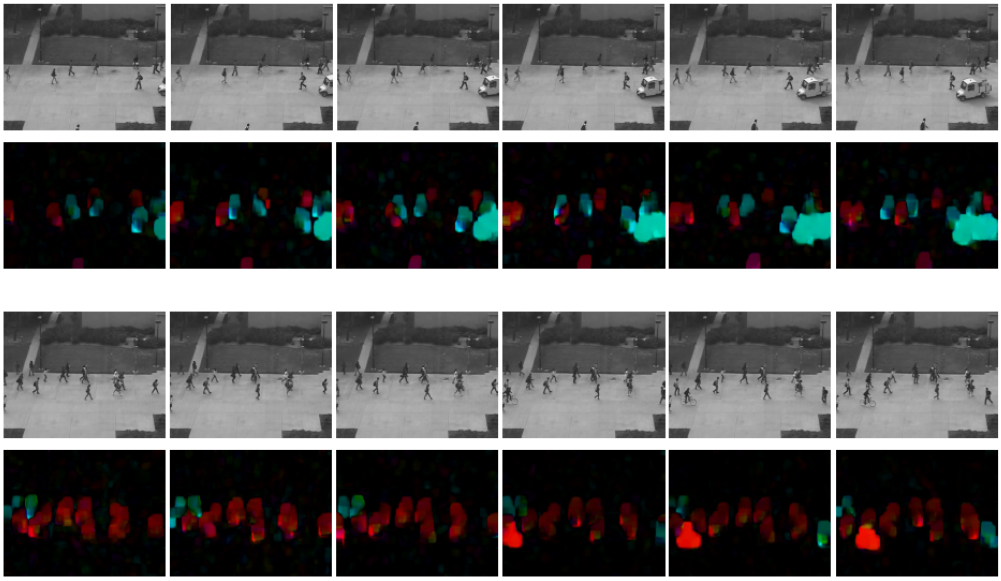


FIGURE 4.4: Optical flow and original images

After training the CAEs, the encoder part of each of them represent the FCNs of our two-stream architecture. For each input frame F_t represented by the video volumes F and OF, each network provides a feature map of dimension 676*256. We combine these two features maps to obtain representation of dimension 676*512, where each row (feature vector) represents a patch of size 27*27 of the original input frame. This architecture allows us, by means of the first FCN, to obtain a robust spatio-temporal representation of each patch of the input frame and refine this representation using the second FCN, which allows a more robust representation of the movement by using the optical flow descriptor. Thanks to this architecture, each small region of the input video volumes is represented by a feature vector able to describe the shapes and the movements contained in this region. In test phase both optical flow and original frames are used and we propose to complete our architecture with a robust Gaussian classifier that allows us to dissociate between the normal and abnormal patches of each frame through the classification of their representative feature vectors. The classification of the feature vectors corresponding to small regions of the input images is carried out according to the following process: in the first step, we extract feature vectors $X = \{x_i\}, x_i \in R^{512}$ from the normal training examples, the mean M and the inverse of the covariance matrix Q of X are then calculated. In the second step, we evaluate each feature vector x_j of the testing frames with Mahalanobis distance d_j using M and Q. This is represented in the following equation:

$$d_j = (x_j - M) * Q * (x_j - M)' \tag{4.24}$$

The outlier vectors, which actually represents abnormal frames, are then picked by thresholding the distance. If the distance exceeds a threshold α, the vector x_j is considered as outlier and the frame p_j is labeled as abnormal, Eq (4.25).

$$p_j : \begin{cases} Normal & if \quad d_j \leq \alpha \\ Abnormal & if \quad d_j > \alpha \end{cases} \tag{4.25}$$

TABLE 4.1: CAEs parameters

Layer	Filters	Kernel (h,w,d)	Stride(h,w,d)
Conv1	64	[11,11,1]	[2,2,1]
Conv2	96	[3,3,1]	[1,1,1]
Conv3	128	[3,3,3]	[2,2,1]
Conv4	256	[3,3,1]	[2,2,1]
Deconv1	256	[3,3,1]	[2,2,1]
Deconv2	128	[3,3,3]	[2,2,1]
Deconv3	96	[3,3,1]	[1,1,1]
Deconv4	1	[11,11,1]	[2,2,1]

4.3.2 TS-FCN 2

The extraction of optical flow images in the test phase allows the system to execute an additional task to extract optical flow images. Moreover, in the training phase the representations of the two volumes of two streams are independents. Then we propose a second method based on new architecture of one block to represent our TS-FCN to rectify the imperfections of the first method. This TS-FCN is learned using normal training samples representations of original images only. It is composed by eight 3D convolution layers (encoder) , eight 3D deconvolution layers (decoder) and one concatenation layer to combine both presentations. The TS-FCN takes as input 3D volumes of three consecutive frames F: $\{F_t; F_{t-1}; F_{t-2}\}$. and try not only to reconstruct those frames but also to reconstruct the optical flow 3D volumes of OF: $\{OF_t; OF_{t-1}; OF_{t-2}\}$ at the same time. The Mean Squared Error is used as loss function to train our model Figure 4.5. After training phase, the encoder part contained 8 convolutions layers represents our TS-FCN. Our model provides a feature map of dimension 676*512 in this case. It is capable to obtain a robust spatio-temporal representation of each both shapes and motion into frames. In test phase, we do not need to extract the optical flow represention manually but our model is capable to construct new representation of optical flow from original frames which are more dedicated to the task of the detection of anomalies Figure 4.6. However, the classification task is done as the same way of the first method.

4.4 EXPERIMENT RESULTS

To evaluate the proposed architecture, we used the USCD Ped2 dataset and compared our results to the state-of-the-art methods. The UCSD Ped2 dataset has 16 folders of training and 12 for testing. The training part of the dataset contains only normal events summarized in pedestrian movements. The testing folders in addition to pedestrians also contain abnormal events that result in the appearance of non-pedestrians. We evaluate our different methods using (Error Equal Rate) EER and (Area Under Curve ROC) AUC as evaluations criteria. A smaller EER corresponds with better performance. As for the AUC, a bigger value corresponds with better performance. The frames and their corresponding optical flow representations are extracted from the raw videos and resized to 227×227. We then subtract

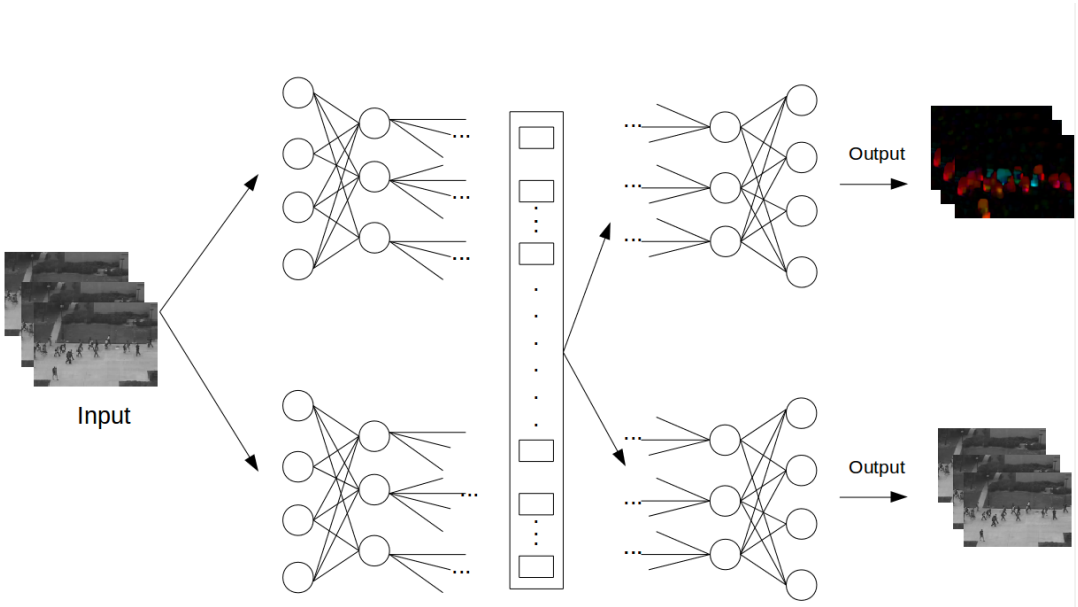


FIGURE 4.5: Our TS-FCN 2 Architecture

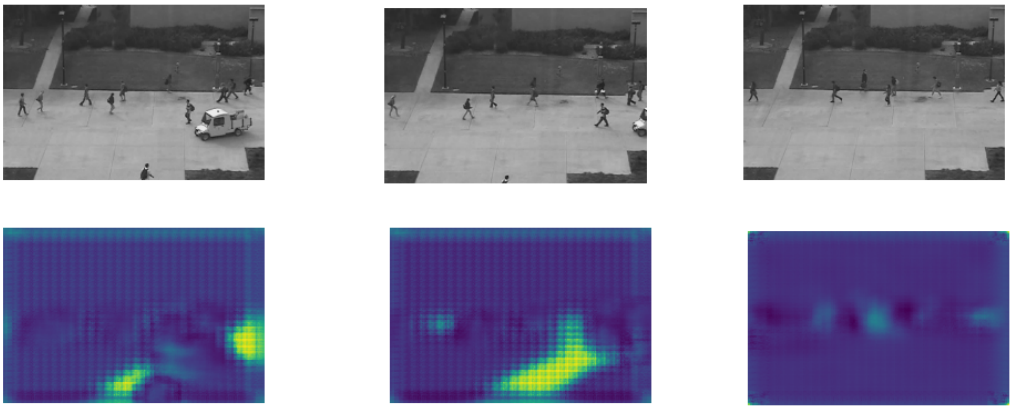


FIGURE 4.6: constructed optical flow in TS-FCN 2

a mean image from each frame contained in the same folder. The mean image is obtained by averaging the frames of each training folder. After the mean subtraction, we scale the pixel values between -1 and 1. For the testing images, we use the mean image calculated during the training to ensure the condition of real world applications. We then group these pre-processed images and the optical flow representations in video volumes composed of 3 consecutive frames. During the training procedure these video volumes are then introduced as inputs to train the two CAE (method 1). We train the two CAE by minimizing the reconstruction error of the input volumes using Adam optimizer. A hyperbolic tangent is used as activation function of each convolution and deconvolution layer to ensure the symmetry of the reconstructed and the input video volumes. The detailed parameters of our network are provided in Table [1]. However only group of pre-processed images is used to train our architecture to reconstruct both pre-processed images and the optical flow representations (method 2). During the testing phase we use only the encoder parts (FCNs), with Gaussian classifier to detect abnormalities in the testing frames. A comparison with state-of-the-art methods are related in Table [2]. We evaluate not only our both methods but also the spatio-temporal

FCN (ST-FCN) individually. These experiments demonstrate the utility of combining the two FCNs as the EER progresses from 19% to 13% (method 1). Which make the importance of using of optical flow image to represent the motion in each frames. Moreover method 2 proves that the coherence of both shapes and motion features in the training phase makes our architecture more robust. It obtained an EER egal to 8.45% and achieves AUC more than 93%. The ROC curve is plotted according to the detection results. The FPR is the rate of incorrectly detected frames to all normal frames in ground truth and the TPR is the rate of correctly detected frames to all abnormal frames in ground truth. We quantify the performance in terms of the equal error rate (EER) and the area under ROC curve (AUC). The EER is the point on the ROC curve that FPR is equal to (1-TPR). Our two-stream fully convolutional networks combined with simple classifier demonstrates good performances, equivalent with state-of-the-art methods for anomaly detection detection.

TABLE 4.2: EER and AUC for frame level comparisons on ped2 dataset

Methods	EER	AUC
PCAD.-S. Pham, 2011	29.20	73.98
CAE(FR)M. Ribeiro, 2017	26.00	81.4
ConvAEM. Hasan, 2016	21.7	90.00
EADHung Vu, 2018	16.47	86.43
ChongChong and Tay, 2017a	12	-
SabokrouM. Sabokrou, 2017	8.2	-
ours		
ST-FCN	19	87.15
TS-FCN 1	**13.2**	**91.6**
TS-FCN 2	**8.45**	**93.6**

4.5 Two-Streams FCN optical flow generating enhanced by deep one class

4.5.1 Over-fitting, Under-fitting, Just Right

In this section, deep learning problems of over-fitting and under-fitting are discussed. After training a deep learning model we are situated in one of three cases ; Over-fitting , Under-fitting, Just Right (Figure 4.7).

Over-fit a model is when the model is trained too well on given dataset which means that the model capable of performing well on the training dataset but does not perform well on the test part of datasets, which can be considered in the context of the bias-variance trade-off. Overfitting during training can be spotted when the error on training data decreases to a very small value but the error on the new data or test data increases to a large value. An overfit model is easily diagnosed by monitoring the performance of the model during training by evaluating it on both a training dataset and on a holdout validation dataset. Under-fit a model when the model fails to sufficiently learn the problem and performs poorly on a training dataset and does not perform well on a holdout sample. An underfit model has high bias and low variance. Regardless of the specific samples in the training data, it cannot learn the problem. An overfit model has low bias and high variance. The model learns the training data too well and performance varies widely with new unseen examples or even statistical noise added to examples in the training dataset. The reason for underfitting can be because of the limited capacity of the network, a limited number of features provided as input to the

network, noisy data etc. Underfitting is not a widely discussed as it is easy to detect the remedy is to try different machine learning algorithm, provide more capacity to a deep neural network, remove noise from the input data, increasing the training time etc. However, the just right or good fit a model, when the model suitably learns the training dataset and generalizes well to the old out dataset.

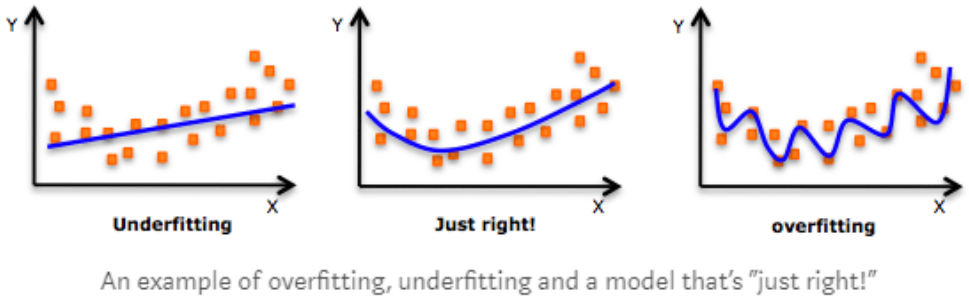


FIGURE 4.7: Deep Learning situations after trained a model

Regularisation

Deep neural networks like CNN are prone to overfitting because of the millions or billions of parameters it encloses. A model with these many parameters can overfit on the training data because it has sufficient capacity to do so.
By removing certain layers or decreasing the number of neurons (filters in CNN) the network becomes less prone to overfitting as the neurons contributing to overfitting are removed or deactivated. The network also has a reduced number of parameters because of which it cannot memorize all the data points will be forced to generalize. There is no general rule as to how many layers are to be removed or how many neurons must be in a layer before the network can overfit. The popular approach for reducing the network complexity is :

- Grid search can be applied to find out the number of neurons and/or layers to reduce or remove overfitting.
- The overfit model can be pruned (trimmed) by removing nodes or connections until it reaches suitable performance on test data.

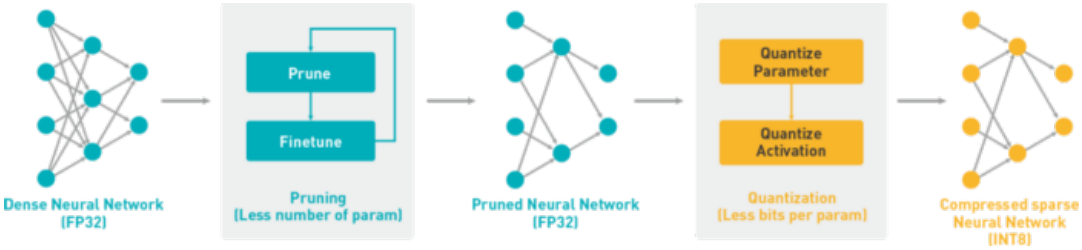


FIGURE 4.8: Deep Learning situations after trained a model

Weight regularization is a technique which aims to stabilize an overfitted network by penalizing the large value of weights in the network. An overfitted network usually presents with problems with a large value of weights as a small change in the input can lead to large changes in the output. For instance, when the network is given new or test data, it results in

incorrect predictions. Weight regularization penalizes the weights of the network and forcing the optimization algorithm to reduce the larger weight values to smaller weights, and this leads to stability of the network and presents good performance. In weight regularization, the network configuration remains unchanged only modifying the value of weights. Weight Regularization reduces overfitting by penalizing or adding a constraint to the loss function. Regularization terms are constraints the optimization algorithm (like Stochastic Gradient Descent) must adhere to when minimizing loss function apart from minimizing the error between predicted value and actual value; for example in L1 regularisation, the term of $\alpha |W_i|$ is added the cost function equation (4.26) When the term of αW_i^2 is added in the case of L2 regularisation Equation 4.27

$$cost_{tot} = cost + \alpha |W_i| \tag{4.26}$$

$$cost_{tot} = cost + \alpha W_i^2 \tag{4.27}$$

By adding a weight penalty to the loss function the overall loss/cost of the network increases. The optimizer will now be forced to minimize the weights of the network as that is contributing more to the overall loss. By increasing the error/loss the error gradient wrt weights increases, which in turn results in a bigger change in weight update. Without the weight penalty, the gradient value remains very small thus the change in weights also remains small. With an increase in error gradient, the large weight values are reduced to a smaller value in the weight update rule. Larger weights result in a larger penalty to the loss function, thus pushing the network towards smaller stabilized weight values. L1 regularization adds the sum of absolute values of the weights in the network as the weight penalty. L2 regularization adds the squared values of weights as the weight penalty.

The lambda term is a hyperparameter which defines how much of the network's weights must be reflected on the loss function or simply the term which controls the influence of weight penalty on the loss function. If the data is too complex, L2 regularization is a better choice as it can model the inherent pattern in the data. If the data is simple, L1 regularization can be used. For most computer vision L2 regularization equation weight decay is applied.

However, there are many other technique for regularisation such Dropout, it is a regularization strategy that prevents deep neural networks from overfitting. While L1 L2 regularization reduces overfitting by modifying the loss function, dropouts, on the other hand, deactivate a certain number of neurons at a layer from firing during training.

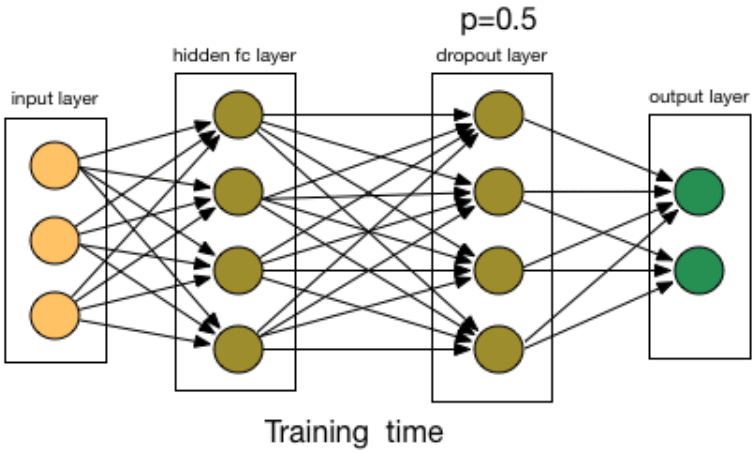


FIGURE 4.9: Dropout

At each iteration different set of neurons are deactivated this results in a different set of results. Many deep learning frameworks implement dropouts as a layer which receives inputs from the previous layer, the dropout layer randomly selects neurons which are not fired to the next layer. By deactivating certain neurons which might contribute to overfitting the performance of the network on test data improves.

Dropouts reduce overfitting in a variety of problems like image classification, image segmentation, word embedding etc.

4.5.2 Experiments results

In this section, we propose end-to-end new unsupervised architecture (Figure 4.7) for anomaly detection in video drone. It is trained with only with normal consecutive original and optical flow frames. Our architecture capable of constructing new optical flow representations of a drone's video from consecutive original images. It is based on a mixture of convolution and deconvolution layers capable of not only generating optical flow images automatically but also extract compact features from both original and optical flow images at the testing phase. It can produce optical flow representations of abnormal samples with greater Error Generation (EG) of Optical Flow compared to the normal samples intuited by the decreasing of the intra-class distance for normal class at the training phase (equation 1).

$$EG = \frac{1}{n}\sum_{1}^{n}(\phi(i) - \hat{\phi}(i))^2 \tag{4.28}$$

When $\phi(i)$ is the original optical flow and $\hat{\phi}(i)$ is the generated optical flow. Thanks to this architecture our model is able to represent properly both shapes and motion in videos. It is composed of 8 convolution layers, one concatenation layer, to combine features maps from each 4 convolution layers, and 8 deconvolution layers to construct the input of consecutive original images and to generate consecutive optical flow. The concatenation layer is our bottleneck layer. We called our architecture a CNN Generator of optical flow due to its capability to generate optical flow samples from original images. The hyper-parameters of our architecture is provided in the following Table 1.

TABLE 4.3: Our architecture hyperparameters

Layer	Filters	Kernel (h,w,d)	Stride(h,w,d)
Conv1	64	[11,11,1]	[2,2,1]
Conv2	128	[3,3,1]	[1,1,1]
Conv3	256	[3,3,3]	[2,2,1]
Conv4	512	[3,3,1]	[2,2,1]
Conv5	64	[11,11,1]	[2,2,1]
Conv6	128	[3,3,1]	[1,1,1]
Conv7	256	[3,3,3]	[2,2,1]
Conv8	512	[3,3,1]	[2,2,1]
Concat	1024	—-	——
Deconv1	512	[3,3,1]	[2,2,1]
Deconv2	256	[3,3,3]	[2,2,1]
Deconv3	128	[3,3,1]	[1,1,1]
Deconv4	1	[11,11,1]	[2,2,1]
Deconv5	512	[3,3,1]	[2,2,1]
Deconv6	256	[3,3,3]	[2,2,1]
Deconv7	128	[3,3,1]	[1,1,1]
Deconv8	1	[11,11,1]	[2,2,1]

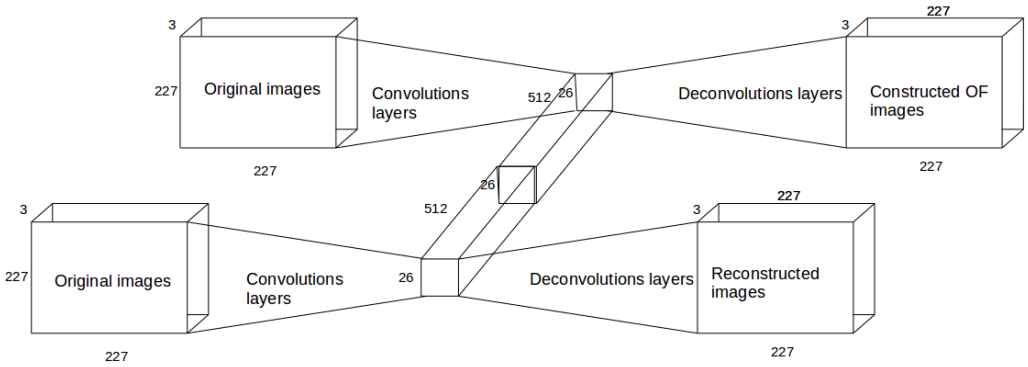


FIGURE 4.10: Our architecture

The Concat represents our concatenation layer, it does not need any filters or any strides as hyper-parameters, it takes Conv4 and Conv8 as inputs.

4.5.3 Training

We propose to train our architecture using only normal samples. We have used as input volumes three consecutive frames F: $\{F_t; F_{t-1}; F_{t-2}\}$ to describe not only the shapes but also the motion between these three frames. Only in the training, the frames and their corresponding optical flow representations are extracted from the raw videos and resized to 227×227. We scale the pixels values in [1,-1]. In the testing phase, we used the same scaling values during the training to ensure the condition of real world applications. Our architecture

is back-propped by Adam optimizer with learning rate equal to 0.00001. Hyperbolic tangent is used as activation function of each convolution and deconvolution layer to ensure the symmetry of the reconstructed and the input video volume. Our architecture is trained with custom loss functions (L) as the sum of three terms as given in equation (12) , compactness loss C_l generation loss G_l and reconstruction loss R_l for training our architectures. The aim of using those three loss is to maximize the inter-classes distance (between normal and abnormal samples) and to minimize the intra-classes distance (between normal samples).

The objective of our G_l loss and C_l loss is to obtain features capable to generalize the training data (Normal samples) and to generate optical flow images with minimum EG. Thus, those terms aim to maximize the inter-classes between normal and abnormal samples. However, the aim of our compactness loss is to obtain compact features of training data by converging both normal features extracted from original images and optical flow images to fixed point C_0. Hence, the compactness aims to minimize the intra-class distance. We fixed the point C_0 as maximum value of our data range which is a vector of ones (Equation 2 and 3).

$$L = \frac{1}{n}(\sum_{i=1}^{n}(V-\hat{V})^2 + \sum_{i=1}^{n}(W-\hat{W})^2) + \alpha\,|M(x_i)-1| \tag{4.29}$$

$$L = R_l + G_l + \alpha C_l \tag{4.30}$$

when, V represents volume of original image and $\hat{V}$ is the corresponded reconstructed volume of V. W is a volume consecutive of the optical flow image and $\hat{w}$ is the corresponded reconstructed volume of W extracted from equation (5). $M(x_i)$ is the mean value of features X_i at each patch in Concat layer. α is a hyperparameter between [0,1[of our custom loss function. It controls the influence the compactness of our features. In practice we fixed α to 0.1 to ensure the scale condition of others terms of L. It should note when α = 0, then the model is training without compactness loss and limited to reconstruction and generation loss.

4.5.4 Testing

After training our architecture, we obtain a model capable of extracting a robust spatio-temporal representation of each patch. Thanks to this architecture, each small region of the input video volumes is represented by a feature vector (26*26*1024) able to describe the shapes and the movements contained in this region.

In test phase, only original frames are used. The optical flow samples are generated by our architecture. We exploit our compactness loss function used at the training for end-to-end anomaly detection to classify our images. The compactness loss is used to compact feature vectors into a half-sphere (S) of center C_0 with radius R : the maximum distance between 1 and the means of training features vectors. Then, of each new video volumes, we extract the mean of features $M'(x_i)$ at Concat layers and compared to the radius R (Equation 4). In order to evaluate our compactness loss we compare our classification with a Mahalanobis distance.

$$\begin{cases} Normal & if \quad 1-M'(x_i) \leq R \\ Abnormal & if \quad 1-M'(x_i) > R \end{cases} \tag{4.31}$$

4.5.5 Minimization of the effect of UAV motion on optical flow images

Optical flow is the pattern of apparent motion of objects between two consecutive frames caused by the movement of object or camera. It is 2D vector field where each vector is a displacement vector showing the movement of points from first frame to second. We use

OpenCV Gunner Farneback's algorithm to extract our dense optical flow. We get a 2-channel array with optical flow vectors, (u,v). The Figure 4.8 shows same samples of optical flow calculated by Farneback's algorithm.

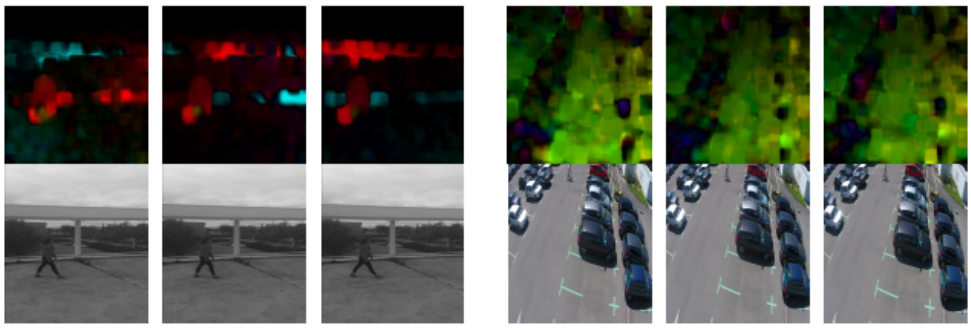


FIGURE 4.11: Optical flow samples of MDVD

In order to denoising and minimize of the effect of UAV motion on optical flow images. We propose to substract the mean optical flow samples in training from the optical flow samples in testing.

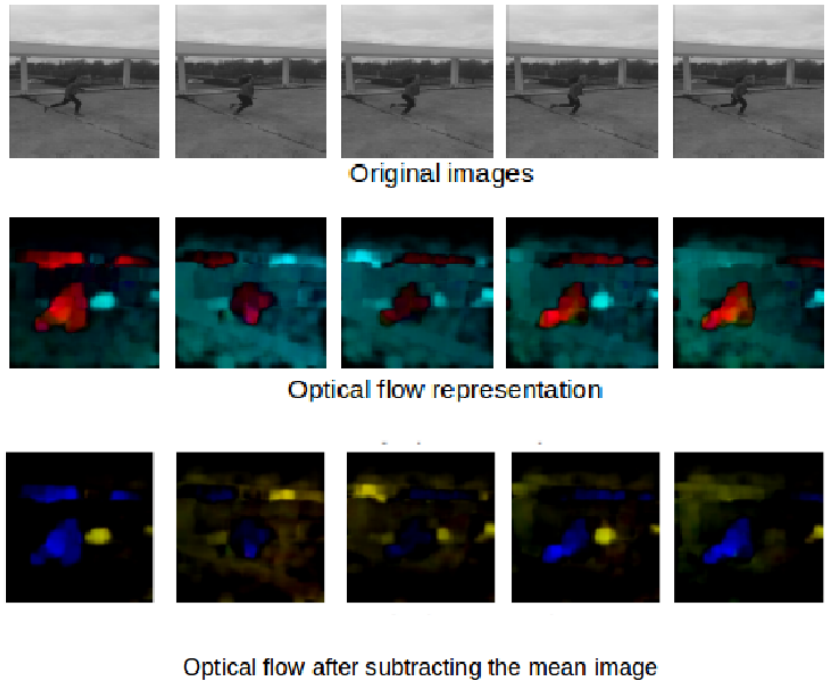


FIGURE 4.12: Subtraction of mean optical flow in other examples

Figure 4.9 and 4.10 show same examples of optical flow of Mini drone dataset and some others examples captured in different scene. These both figures proves that subtracted mean drone motion can minimize the drone motion effect on optical flow frames. Then, these new representations of optical flow are less affected by the movement of the drone and they are less noisy. We have used this version of optical flow to train our architecture.

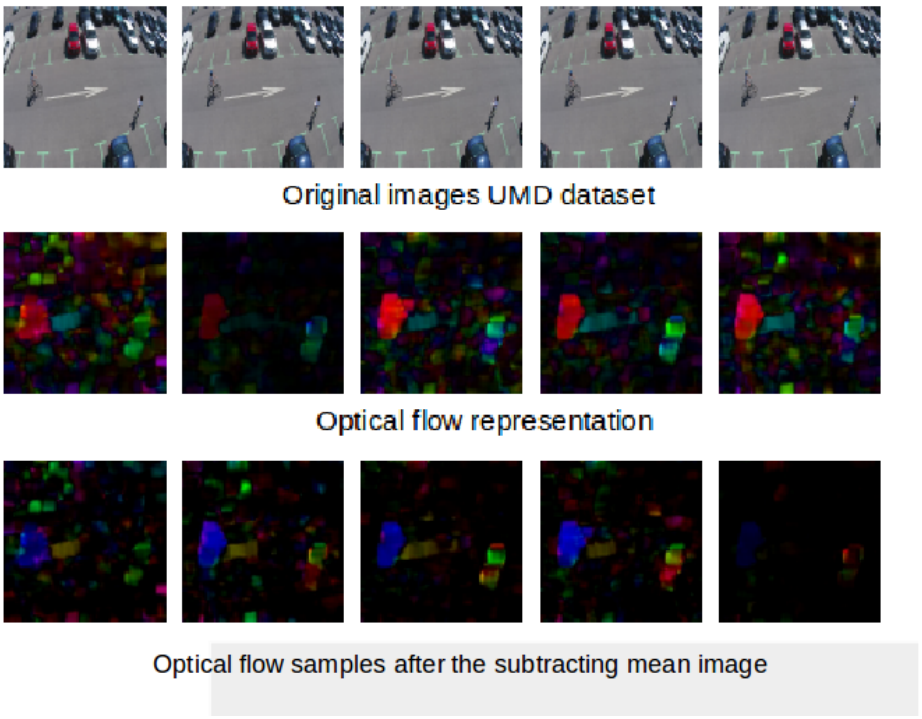


FIGURE 4.13: Subtraction of mean optical flow in MDVD dataset

4.5.6 Experiments results

To evaluate the proposed architecture, we mainly use the Mini-Drone Video Dataset (MDVD) filmed by UAV Bonetto et al., 2015. It is filmed by a drone with type Phantom 2 in a car parking. It is mainly used for events identification. It is composed of 38 videos captured in high resolution, with a duration up to 24 seconds each. The videos in MDVD are divided into three cases: normal, suspicious, and abnormal and they are defined by the actions of the persons implicated in the videos. The normal cases is defined by several events such as people walking, getting in their cars or parking correctly. The Abnormal cases are represented by people fighting, or stealing. Finally for suspicious cases nothing is wrong in real, but people have a suspicious behavior which could distract the attention of the surveillance staff. In order to use MDVD dataset in unsupervised mode for anomaly detection we split this dataset into : 10 videos for the training contains only normal samples and 10 videos for the test contain both abnormal and normal events are used. In other hand, we have tested our method on small own dataset, the normal event when the girl is walking and the abnormal event when she is running, this kind of anomaly are largely used in anomaly detection by fix cameras Patil and Biswas, 2016. We have used Error Equal Rate (EER) and Area Under Curve ROC (AUC) as evaluations criteria. A smaller EER corresponds to better performance. As for the AUC, a bigger value corresponds to better performance .

TABLE 4.4: EER and AUC for frame level comparisons on ped2 dataset

Methods	EER	AUC
Henrio and Nakashima, 2018 [VGG+LSTM]	–	72.75
Henrio and Nakashima, 2018[VGG]	–	50.12
Ours	19.85	85.3

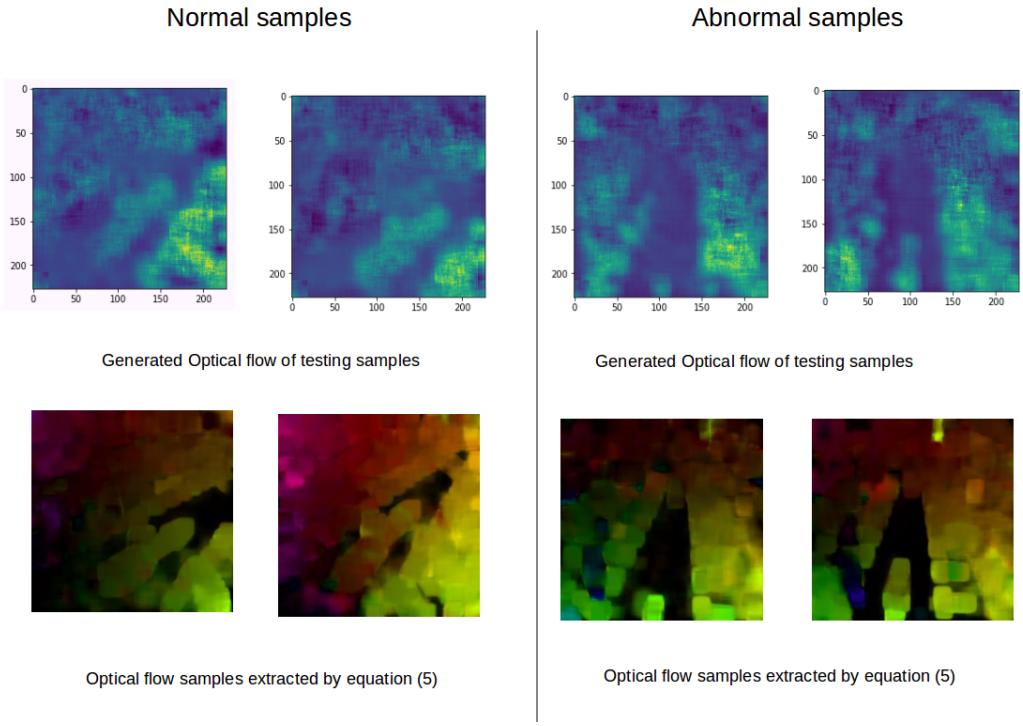


FIGURE 4.14: samples optical flow generated by our architecture

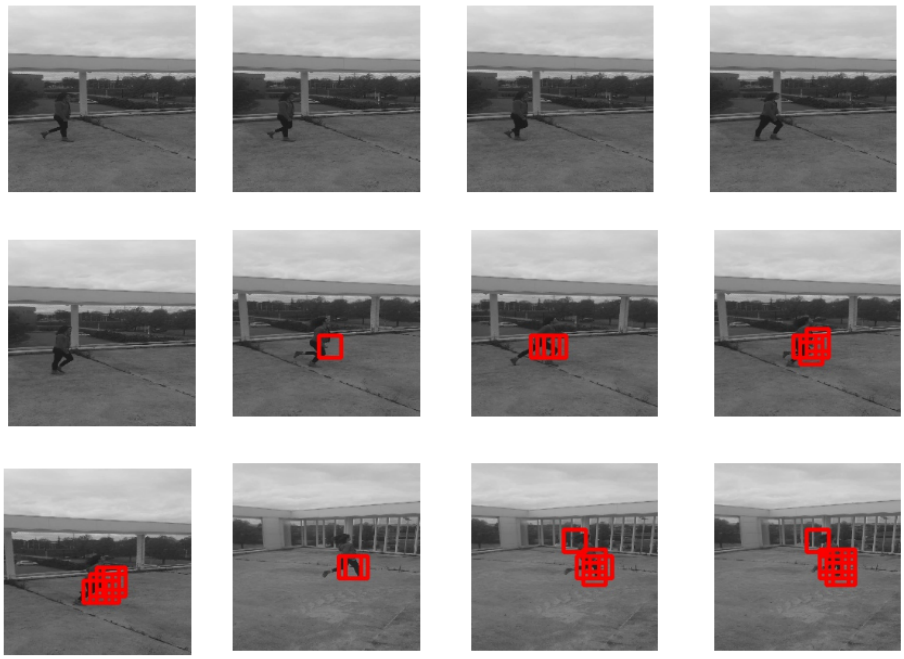


FIGURE 4.15: Our results on others examples

The Figure 4.11 shows the generated optical flow frames in MDVD and the Figure 4.13 shows our results on MDVD, it proves that our method can localize the anomaly which is the biker or fighting in this figure. But when the motion of drone is fast our system can give same error of localisation, but it still can dissociate between abnormal and normal at frame level.

Despite the difference between the movements and trajectories of the drone in the training phase and the testing phase. The results prove that our architecture works properly and it is robust to detect abnormal events and shapes.

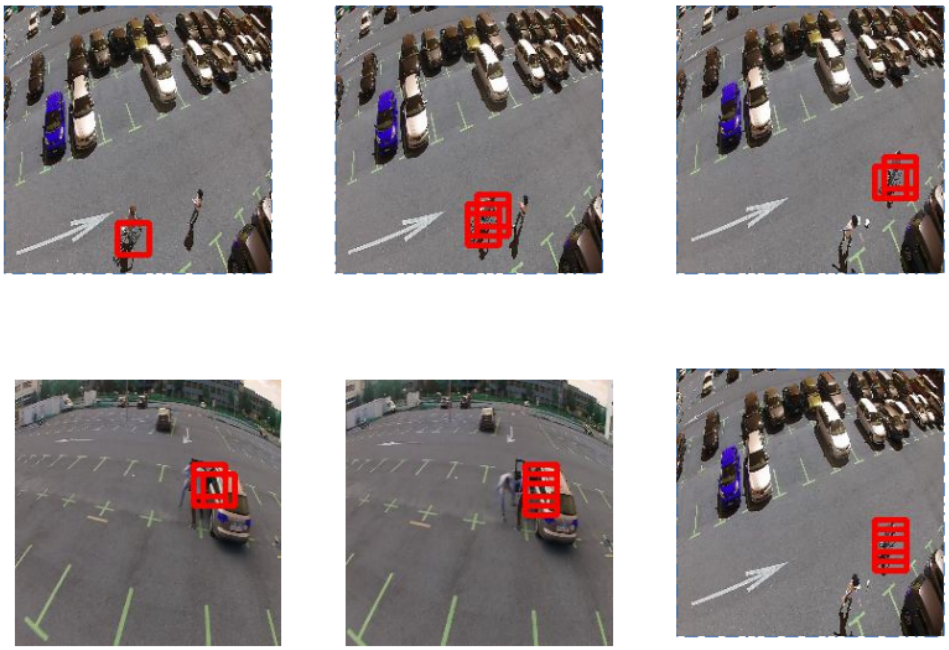


FIGURE 4.16: Our results on MDVD dataset

The Figure 4.12 represents our results on own dataset. It shows that our method capable of detect abnormal brutal movement (Running in this case)

In order to evaluate our compactness loss. We trained our model without compactness loss. The Table 5 shows our results on MDVD using Mahalanobis distance (Equation 5).

$$D = (y_j - M) * Q * (y_j - M)' Mahalanobisdistance : \begin{cases} Normal & if \quad D \leq \alpha \\ Abnormal & if \quad D > \alpha \end{cases} \tag{4.32}$$

When M is the mean and Q is the inverse of the covariance matrix of the training data X. If the distance exceeds a threshold α, the testing vector y_j is considered as outlier and the corresponded frame is labeled as abnormal.

TABLE 4.5: Compactness loss importance)

–	EER	AUC
our(Without compactness)	23	78.2
our(With compactness)	19.85	85.3

The TABLE 3 shows our results on MDVD data sets. It proves that it has more efficient results compared to Mahalanobis classifier.

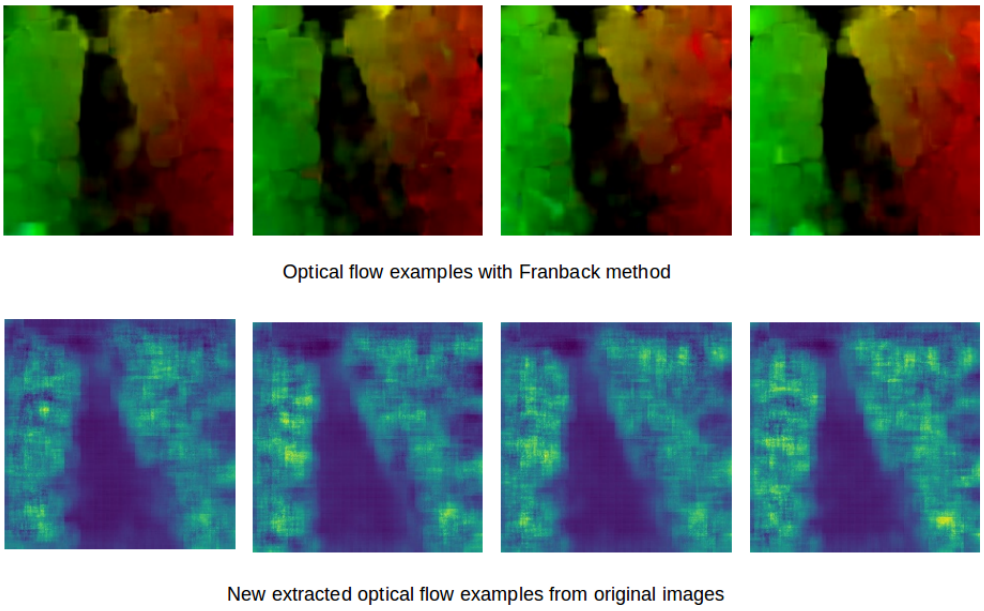


FIGURE 4.17: Samples of generated Optical flow

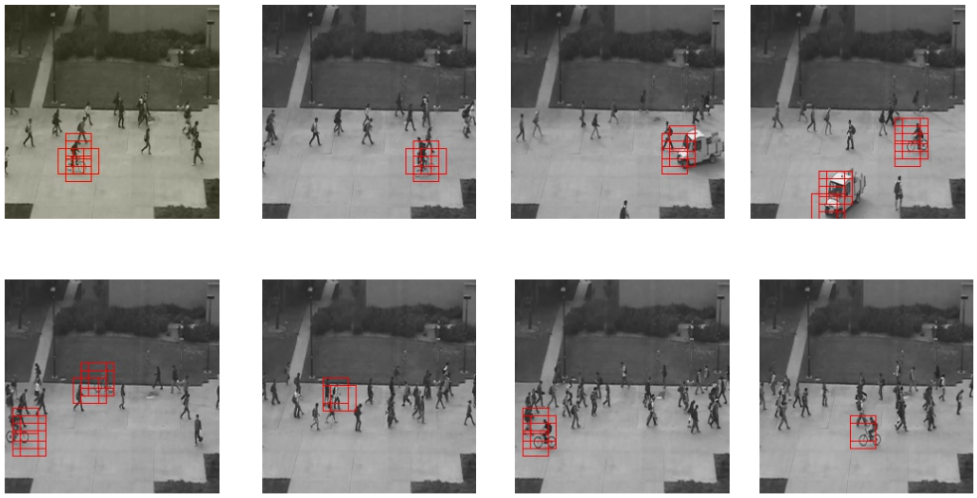


FIGURE 4.18: Our results on Ped2 dataset

The Figure 4.14 shows some samples of generated optical flow of video drone. It's proves the performance of in term of generating optical flow at the testing phase.

In order to compare the capacity of our architecture to other recent researches in the videos anomaly detection. We have tested our method on data set with fixed cameras. We trained our model on UCSD Ped2 dataset and compared our results to the state-of-the-art methods.

The UCSD Ped2 dataset has 16 folders of training and 12 for testing. The training folders of this data set contains only normal events (pedestrian walking). The testing folders contains both normal and abnormal samples such as the appearance of non-pedestrians in pedestrians

area. The TABLE 4 and the Figure 4.15 show the robustness and the efficiency of our method in video anomaly detection.

TABLE 4.6: EER and AUC for frame level comparisons on Ped2 dataset

Methods	EER	AUC
Mehran, Oyama, and Shah, 2009b	40	-
Kim and Grauman, 2009b	30.71	-
Pham et al., 2011	29.20	73.98
Ribeiro, Lazzaretti, and Lopes, 2018	26.00	81.4
Hamdi et al., 2019	14.50	-
Sabokrou et al., 2017	8.2	-
ours	8.1	94.9

4.6 UTT Drone dataset

In this section we propose new dataset called UTT drone dataset, it is captured by Mavic air 2 from DJI series. It is composed of 7 folders for the train and 12 folders for the test. The train folders contains only normal events like : people are walking on lawn and the test folders contain both normal and abnormal events ; people are running, fighting, or falling down (Figure 4.19, 4.20)

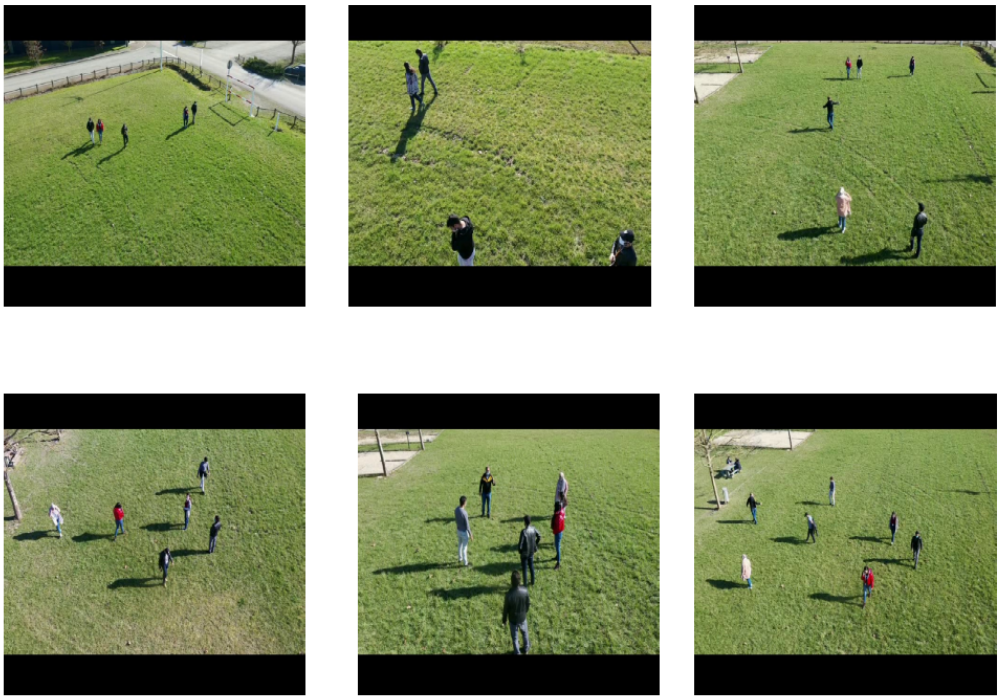


FIGURE 4.19: **Normal event: people are walking**

FIGURE 4.20: Abnormal events ; people are running, fighting, or falling down

Our new datasets have more samples for training compared to the existing datasets for anomaly detection table 4.7. Which make the use of UTT drone dataset more efficient and large enough to train deep models.

TABLE 4.7: Comparison

Data set	Nb of frames at the train	Nb of frames at the test
UCSD Ped2	2 550	2 010
UMN	5 941	7 749
Mini-drone	5 155	6 323
UTT drone	8 933	5 088

Moreover, we have tested our method on this new datasets. Our method have a good performance of anomaly detection (Figure 4.21, 4.22,).

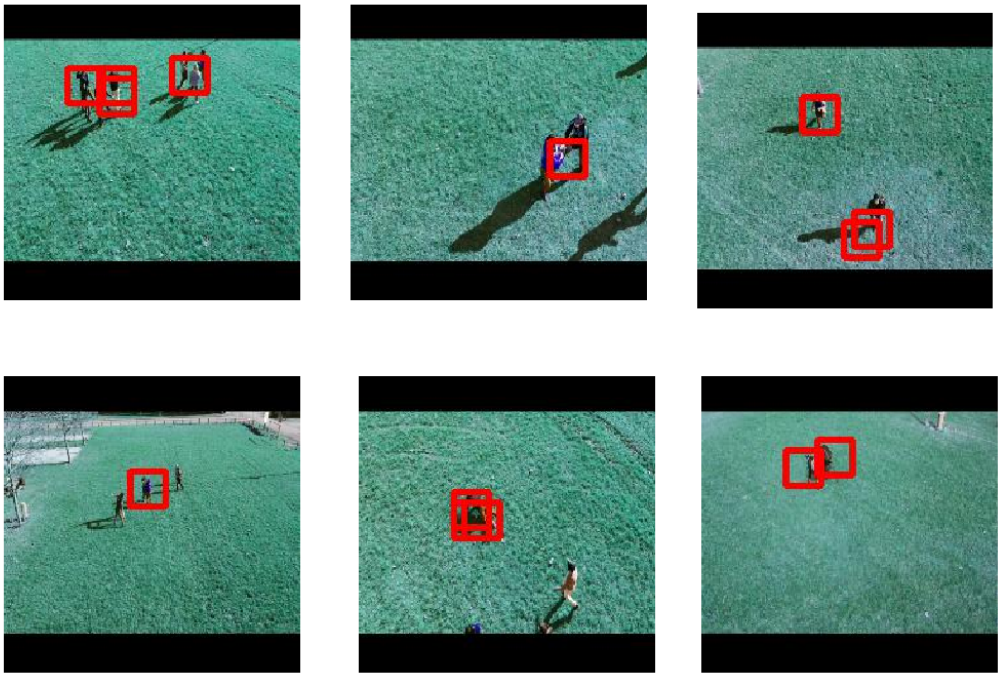


FIGURE 4.21: Our results on UTT drone dataset

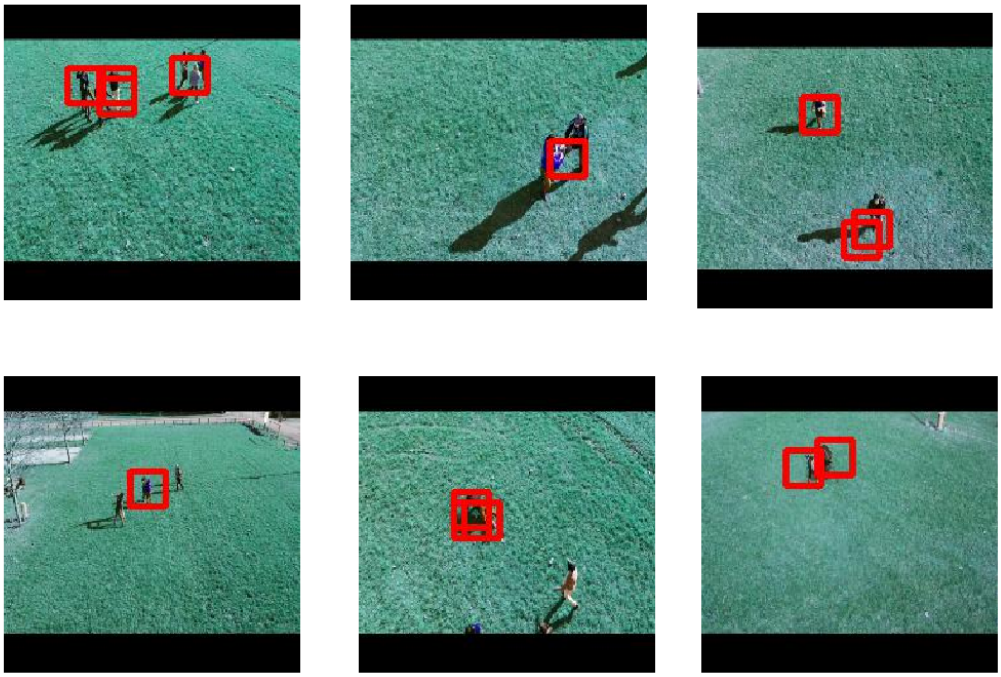


FIGURE 4.22: Our results on UTT drone dataset

4.7 Conclusion

In the first part of this chapter, we proposed a new unsupervised architecture based on a 3D convolutional auto-encoder for spatiotemporal feature map extraction, we have combined this architecture with a classifier exploiting the Mahalanobis distance. We have demonstrated the interest of unsupervised 3D neural networks for the detection and localization of abnormal video events. We have made a two-stream network exploiting both the images and their representations in terms of optical flow. This network has proven to be able to obtain robust and exploitable motion descriptors for the analysis of video events. Thanks to this two-stream architecture, we have seen a clear improvement in terms of qualitative and quantitative results obtained on the UCSD Ped2 dataset. This improvement can be explained by the reinforcement of the representations by motion descriptors extracted thanks to the second network exploiting the optical flow. In the second part, we proposed a new end-to-end method of detection and localization of anomalies. The method is based on a one class 3D neural network trained with an original objective function. The network is trained only on training samples from the normal class. Thanks to the proposed objective function, the network is able to extract robust spatiotemporal representations and ensure the compactness of representations belonging to the normal class. Compared to the state of the art, our method is among the best performing methods on the UCSD Ped2 and Mini-drone dataset.

Chapter 5

Conclusions and Future work

The majority of our work presented in this thesis is based on convolutional neural networks. However, the design of a neural network is not an easy task. There are many choices that affect the performance of the network. These include how to sample and pre-process the input data, the number of layers, their types and the different parameters to be applied to them, the optimizer to be used for training the network and its parameters, the length of the temporal sequence to be used..ect. In addition to the number of parameters, learning a network is not only expensive in terms of hardware resources (GPUs), but also time consuming. The different parameters of a network are closely related to the training data, which means that for different datasets, these parameters are quite different. This aspect, added to the difficulties of design and training of neural networks, forced us to limit the test datasets for our different methods. In our future work, we plan to explore other datasets to confirm the accuracy and relevance of our approaches. Using only normal samples at training can limit the performance of the deep one class classification because there is not abnormal samples used to increase the inter-distance between normal and abnormal features. For that reason we need more samples from both classes. In our future work, we investigate on Deep Fake for example , Art Transfer learning to generate abnormal samples from normal events Figure 5.1 and Figure 5.2. These new samples can be fitted on Siamese Network with deep one class concept to separate abnormal features from normal ones(See Figure 5.3), nor only to increase the inter-distance but also to decrease the intra-distance within normal samples. The figure 5.4 illustrate an example for training Siamese network with both normal sample and abnormal samples. The EER on Ped2 datasets is around 10% but this value can be improved by added Optical flow frames.
To conclude :

- Currently, our unsupervised networks are trained with reconstruction , compactness and generative loss, by added Siamese Network and deep fake can obtain more robust descriptor for movement.
- In the fourth chapter we proposed a two-stream network exploiting optical flow representations to obtain robust temporal representations, However, the extraction of the optical flow can be time consuming, We therefore propose to replace the optical flow by motion vectors that are naturally present in the video and thus directly accessible

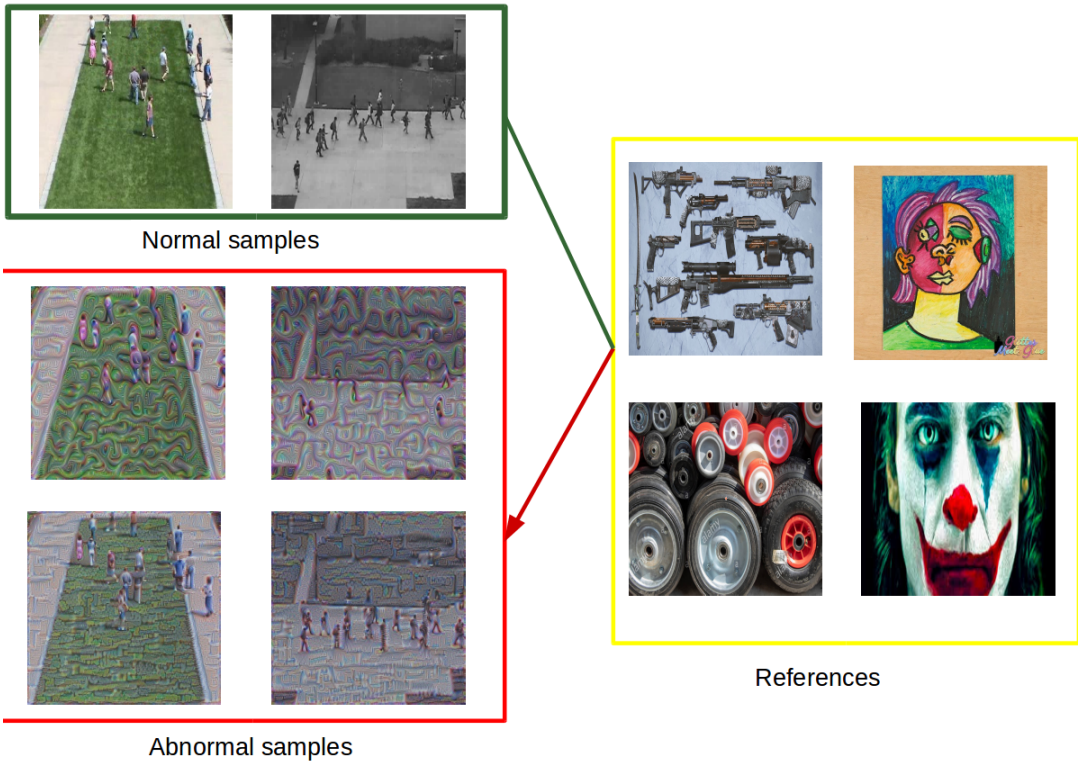


FIGURE 5.1: Art transfer

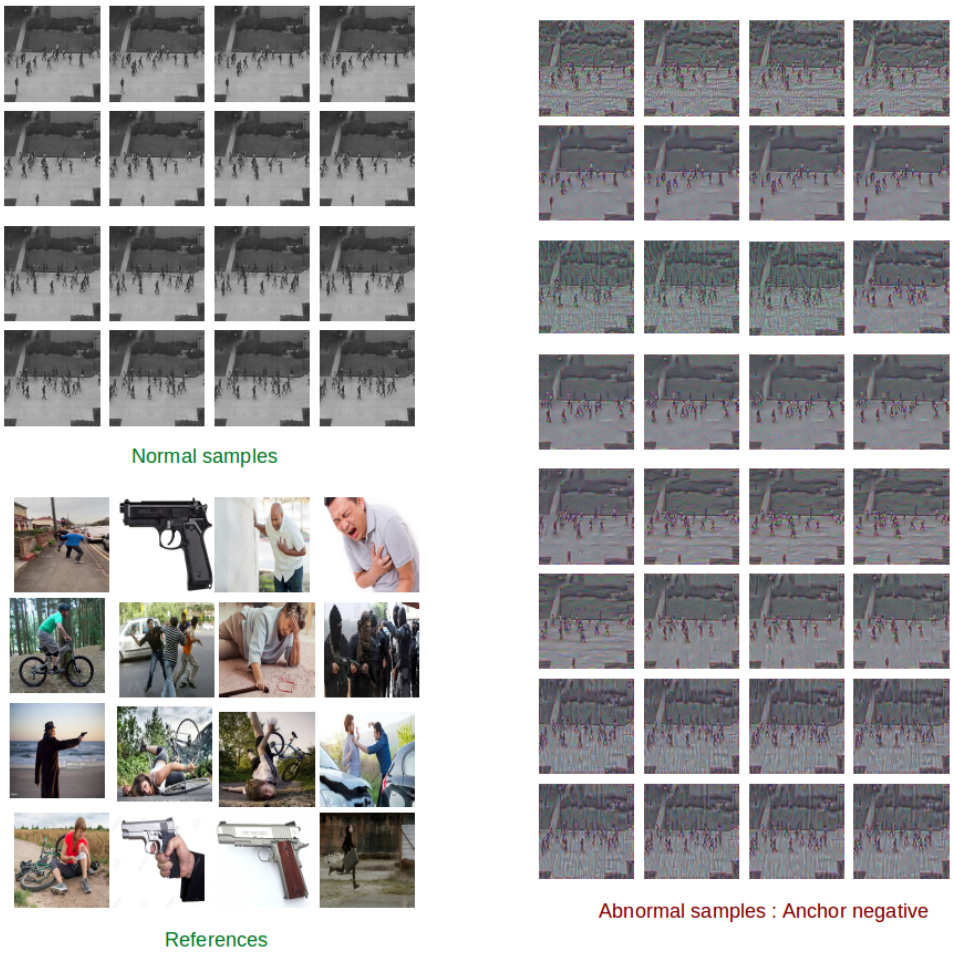


FIGURE 5.2: art transfer

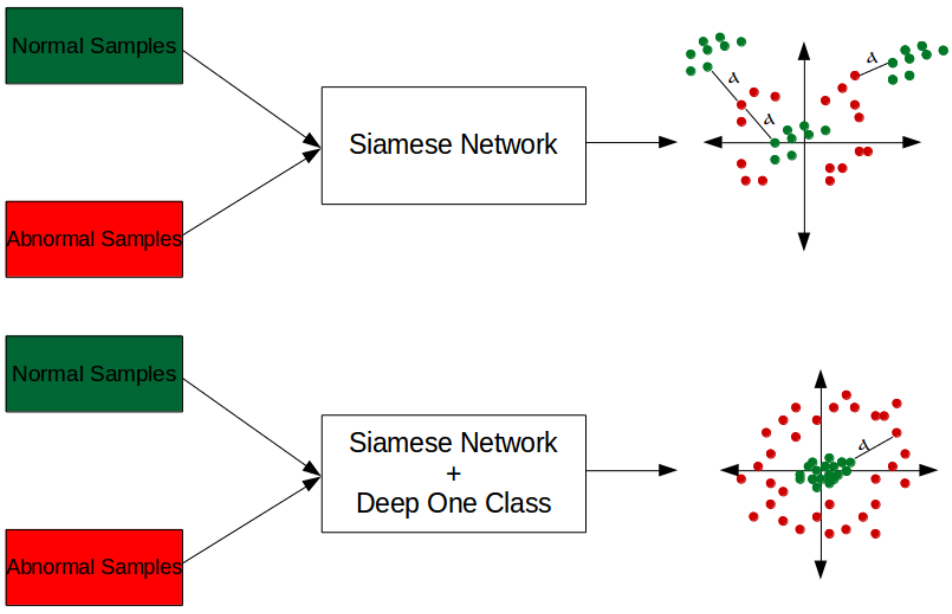


FIGURE 5.3: Deep One class

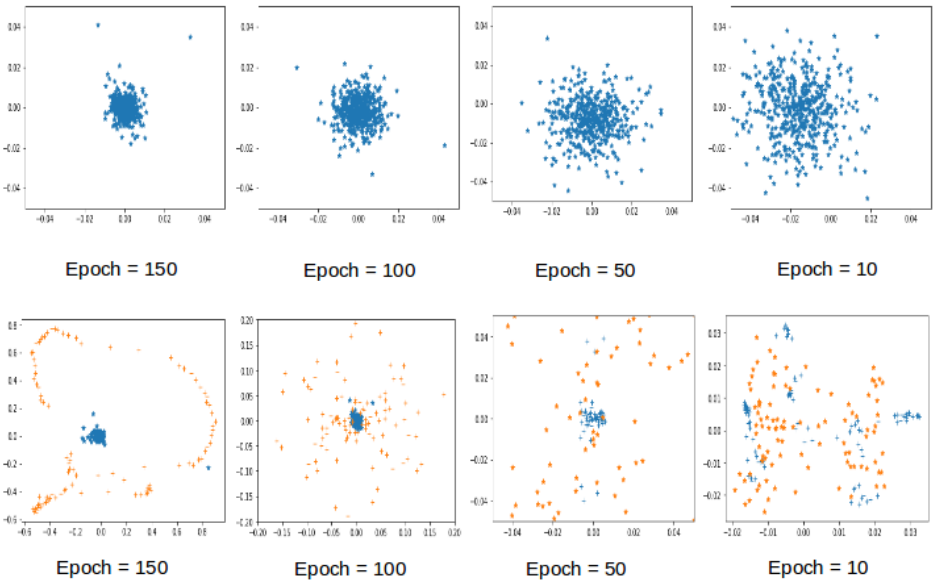


FIGURE 5.4: Deep One class

Chapter 6

Résumé en Français

6.1 Introduction génerale

La sécurité civile est l'ensemble des moyens mis en œuvre par un État ou une organisation pour protéger des populations civiles (personnes morales et physiques), ainsi que leurs biens et activités, en temps de guerre, en temps de crise comme en temps de paix contre des risques et menaces de toute nature, civile ou militaire. Elle consiste notamment à garantir la sécurité des personnes morales et physiques contre les tous type de risques naturels comme incendies d'origine naturelle ou contre les menaces diverses qui peuvent mettre en danger leur sécurité, comme celle de leurs biens ou de leurs activités (actes de terrorisme, actes de vandalisme...etc.). A fin de réprimer et d'endiguer les comportements et les actes « délictueux » sur l'espace public et grâce à son faible coût du matériel, la vidéosurveillance s'impose depuis quelques années maintenant comme l'un des moyens à disposition des municipalités pour garantir la surveillance et la sécurité des biens et des personnes.

Le vidéoprotection est un système de télévision en circuit fermé décrit toute une gamme de technologies de vidéosurveillance. Il s'agit d'un système reliant une ou plusieurs caméras vidéo en circuit fermé ou en boucle; les images captées sont envoyées à un écran central de télévision ou peuvent être traitées automatiquement et/ou visionnées puis archivées ou détruites. Ces types de systèmes sont largement utilisés dans de nombreuses applications telles que les établissements de soins infirmiers, l'application de la loi, la sécurité des bâtiments, et l'analyse des tracés. D'autre part, la nécessité d'un contrôle efficace des lieux tels que les aéroports, les gares, les centres commerciaux, les salles de sport bondées, les installations militaires est en augmentation. En particulier, l'utilisation de drones caméras est devenue très importante dans le domaine de la détection des comportements anormaux. Cette valeur importante vient non seulement du fait qu'un drone peut surveiller des zones très difficiles accessibles par l'être humain, mais aussi du fait qu'il est rentable car il peut remplacer toute une installation de caméras fixes.

Les systèmes de surveillance traditionnels s'appuyaient sur des caméras réseau surveillées par un opérateur humain qui doit être conscient des actions menées par les personnes qui se trouvent dans la scène surveillée. Avec l'augmentation récente du nombre de caméras à analyser, l'efficacité et la précision des opérateurs humains ont atteint une limite. Par exemple, Dans **2**, l'auteur prouve qu'un opérateur peut manquer 60% de cibler les événements lorsqu'elle est chargée de visionner 9 vidéos ou plus. Par la suite le traitement de cette masse importante de données avec les systèmes traditionnels est très difficiles à gérer et parfois quasiment impossible à réaliser ses objectives. De plus, ils sont coûteux et nécessite un nombre important des être-humains pour la surveillance.

6.1.1 Objectives

Ces dernières années, l'apprentissage automatique (AA) a montré sa efficacité considérable dans plusieurs domaines en particulier dans le domaine de la vison par ordinateur. Le Deep Learning (DL) est un ensemble de méthodes d'apprentissage automatique tentant de modéliser avec un haut niveau d'abstraction des données grâce à des architectures articulées de différentes transformations non linéaires. Le DL a permis de rendre envisageable certaines tâches pour l'ordinateur qu'on a pensé jusque-là qu'elle est exclusif à l'homme. Par exemple, classification d'images, reconnaissance faciale , estimation de la pose humaine, traitement automatique du langage naturel , reconnaissance automatique de la parole . Mais aussi d'autre tâches plus complexe comme: systèmes de traduction automatique Chung et al., 2016, lecture labiale Toshev and Szegedy, 2013 mener des négociations Conneau et al., 2016, raisonnement visuel Amodei et al., 2015 et génération automatique de code informatique .

L'apprentissage profond DL (Deep Learning) est un ensemble de méthodes basé sur des architecture à plusieurs couche d'apprentissage de représentation. Ces méthodes permettent d'extraire automatiquement à partir de données brutes les représentations nécessaires à la classification de ces données. Actuellement les méthodes d'apprentissage profond les plus efficaces se basent sur un apprentissage supervisé, de grandes bases de données labellisées, contenant des exemples des différentes classes doivent être utilisé.

C'est dans ce contexte particulièrement complexe que se positionne cette thèse. Nous allons explorer les diverses solutions a fin d'exploiter le potentiel de l'apprentissage profond pour la détection d'événement anormaux dans les flux vidéo drone.

6.1.2 Contributions

Les contribution de cette thèse sont en conformité avec l'objectif principal qui est l'adaptation et le développement de nouvelles architectures basées sur l'apprentissage profond pour la détection d'événements anormaux par une caméra drone. Ces contributions s'articulent autour des trois points suivants:

- Nous définissons des nouvelles bases de données pour la détection des événements anormaux par des caméras de drone pour un apprentissage automatique non supervisé. Pour ces bases de données, seuls des échantillons normaux peuvent être utilisés pendant l'entraînement.

-Nous proposons un modèle hybride pour la détection des événements anormaux par une caméras drone. Le système proposé est un système d'interaction entre deux architectures différentes à fin de résoudre le problème envisagé. La première architecture est un RNN pré-entraîné (réseau de neurones à convolution) et la deuxième est un Histogramme du flux optiques. Les réseaux de neurones convolutifs pré-entraîner ont démontré leur efficacité non seulement dans le domaine sur les quels sont entraînées mais aussi dans d'autres domaines d'applications, cette technique s'appelle le transfert learning. D'autre part, l'utilisation d'images du flux optique à fin de représenter les mouvements se produisent dans un vidéo sont largement répandu comme un descripteur de mouvement. Ce descripteur conserve les informations du flux optique (orientation et magnitude) en les distribuant dans un histogramme (HOF). Un SVM de classe unique est entraîné avec des caractéristiques spatiales pour une classification robuste des formes anormales. De plus, nous proposons une fonction de décision est appliquée pour corriger les fausses alertes et les détections erronées. Avec cette architecture on peut conclure que deux point importants; l'utilisation des images des flux optiques ont montré leurs efficacités pour la description convenable des mouvements dans une scène surveillé et l'utilisation d'une architecture profonde peut mieux servir et décrire les

caractéristiques des images ou un vidéo. D'autre part, la combinaison de deux architectures fait pour différents applications est possible et efficace dans le cas de la détection d'anomalies par drone.

-En tenant compte des avantages du système d'hybride qu'on appliqué. Nous proposons une nouvelle architecture RNN (Appelé RNN générateur du flux optique) ou bien (CNN Générateur). Cette architecture est capable de générer les images du flux optique à partir d'images originales et d'extraire des caractéristiques pour la détection des anomalies. Nous renforçons notre (RNN-Générateur) avec un classificateur gaussien afin de détecter les événements spatio-temporels anormaux qui pourraient présenter un risque pour la sécurité. D'autre part, nous proposons une étude approfondie sur la relation entre les trajectoires de la drone dans une mission de surveillance et les présentations des flux optiques à fin de minimiser l'effet des mouvements de la drone sur ce dernier.

- Pour compléter l'architecture du RNN Générateur. Nous proposons une nouvelle de bout en bout architecture pour l'apprentissage en classe unique pour la détection des anomalies. Notre méthode consiste à entraîner le modèle RNN Générateur avec une fonction de perte personnalisée comme la somme de trois termes, la perte de reconstruction (Rl), la perte de générateur (Gl) et la perte de compacité (Cl). L'objectif de la perte de compacité est de compacter les caractéristiques d'entraînement (classe normale) à une semi-hypersphère (SHS).

6.2 Etat de l'art

L'utilisation des drones est en plein essor dans le monde entier, avec une grande variété d'applications potentielles : mise en réseau acoustique sans fil pour la surveillance des drones amateurs, mise à jour de la mise en réseau des drones à l'aide de radios définies par logiciel (SDR) et de réseaux définis par logiciel (SDN), cadre d'apprentissage par renforcement multi-agent (MARL) et détection des points d'accès Wi-Fi malveillants. En particulier, l'utilisation de la caméra du drone est devenue très importante dans le domaine de la détection des comportements anormaux dans les séquences vidéo. Cette importance découle du fait que non seulement un drone peut surveiller des zones étendues et dangereuses, mais aussi qu'il est rentable et peut remplacer toute une installation de caméras fixes Henrio and Nakashima, 2018. En outre, le traitement de séquences vidéo provenant de drones pour la détection d'anomalies est une tâche complexe par rapport à son homologue avec des caméras fixes pour deux raisons : (a) le manque d'ensembles de données vidéo provenant de drones en conditions réelles et (b) les arrière-plans dynamiques, à luminosité variable et à grande échelle. Un système de vidéoprotection par drone est un système de télévision en circuit fermé (CCTV) qui décrit toute une gamme de technologies de vidéosurveillance. De nombreux facteurs peuvent réduire considérablement l'efficacité des systèmes CCTV, par exemple : la fatigue et la lassitude causées par le visionnage prolongé de nombreuses vidéos de surveillance. Une solution possible à ce problème serait l'utilisation de systèmes de vidéosurveillance intelligents. Ces systèmes doivent être capables d'analyser et de modéliser le comportement normal d'une scène surveillée et de détecter tout comportement anormal qui pourrait représenter un risque pour la sécurité. Ces dernières années, des avancées technologiques considérables dans les domaines de l'apprentissage automatique et de la vision par ordinateur ont permis de traiter les systèmes CCTV. Certaines d'entre elles sont des classiques de l'apprentissage automatique : classification d'images, He et al., 2015, reconnaissance faciale, Taigman et al., 2014, estimation de la pose humaine, Toshev and Szegedy, 2013, traitement du langage

naturel, Conneau et al., 2016, reconnaissance vocale automatique, Amodei et al., 2015, et même des tâches plus atypiques ; systèmes de traduction automatique, lecture labiale, Chung et al., 2016 et génération automatique de code logiciel, Lao, Han, and De With, 2009. De plus, l'apprentissage profond (Deep Learning, DL) est un sous-domaine de l'apprentissage automatique (Machine Learning, ML), il vise à apprendre des abstractions de haut niveau dans les données en utilisant des architectures à plusieurs niveaux. Ces différents niveaux sont obtenus en empilant plusieurs modules de transformation non linéaires. Chaque module transforme les données à un niveau différent jusqu'à obtenir une représentation adéquate pour réaliser la tâche cible. L'apprentissage profond a permis de dépasser le modèle traditionnel dans certains cas d'application et de concevoir des systèmes de reconnaissance de formes efficaces sans expertise approfondie des éléments cibles. En fait, les méthodes d'apprentissage profond les plus efficaces sont basées sur l'apprentissage supervisé, en utilisant de grandes bases de données étiquetées contenant des échantillons de différentes classes. Pour tirer parti de ces matériaux d'apprentissage dans un système de surveillance intelligent, une grande quantité de données d'apprentissage représentatives d'événements normaux et anormaux est nécessaire. Les événements anormaux sont les événements rares qui n'apparaissent pas de manière redondante sur la scène. Ainsi, il existe de nombreux obstacles à la création de telles bases de données, nous pouvons par exemple citer les suivants :

- L'aspect contextuel de l'événement. En effet, un événement est fortement lié à son contexte, un événement anormal dans une scène peut être normal dans une autre. Ce point rend presque impossible la conception de bases de données communes qui peuvent être utilisées uniformément pour différentes scènes.

- Les risques et la variabilité de la reproduction de certains événements anormaux font qu'il est impossible d'identifier et de générer suffisamment d'échantillons d'entraînement.

Les événements vidéo anormaux ont été désignés par de nombreux noms dans la littérature, tels qu'anomalie, comportement irrégulier, comportement inhabituel, ou comportement anormal, etc. Ces différents noms seront utilisés alternativement sans se soucier de l'incohérence technique. La détection d'événements vidéo anormaux est également caractérisée par une variété de stratégies de traitement des données d'entraînement. Une première approche consiste à effectuer l'entraînement uniquement sur des données normales et à considérer tout type d'événement en dehors de la phase d'entraînement comme anormal. Une autre approche, à l'opposé de la première, consiste à n'utiliser que les événements anormaux pour la apprentissage, Zhang et al., 2010. Cette approche peut être efficace pour identifier un certain type d'événements anormaux, mais présente un risque élevé de manquer des événements anormaux différents de ceux qui ont été formés. Une autre approche est basée sur l'utilisation de données étiquetées dans deux classes différentes, normale et anormale, Zhou et al., 2016b. D'autres travaux utilisent des données classées et étiquetées plus avancées, où chaque classe représente un type d'événement spécifique, Lao, Han, and De With, 2009. Les approches qui utilisent des événements anormaux comme données d'apprentissage ont souvent des limites. Certains événements anormaux sont impossibles à reproduire. La variabilité des événements anormaux complique grandement la tâche d'apprentissage et peut avoir un effet négatif sur la modélisation. D'autres approches sont basées sur des méthodes de regroupement avec l'utilisation de bases de données non étiquetées contenant à la fois des données normales et anormales, Roshtkhari and Levine, 2013. On suppose que les événements normaux sont ceux qui se produisent fréquemment et que les événements anormaux sont ceux qui se produisent rarement. L'avantage de cette approche est qu'elle ne nécessite aucun

étiquetage des données d'entraînement, mais son efficacité est compromise par l'hypothèse selon laquelle tous les événements rares sont anormaux car, de toute évidence, un événement rare n'est pas nécessairement anormal. Malgré les différentes stratégies de apprentissage de données sur la détection d'événements anormaux, Hasan et al., 2016, Lee, Kim, and Ro, 2018, Oza and Patel, 2019a. La première approche, qui consiste à n'utiliser que des données normales pendant la phase d'apprentissage, est devenue la norme. Dans notre travail, nous adoptons cette approche et nous proposons une nouvelle architecture capable de détecter des événements anormaux en s'entraînant uniquement avec des échantillons normaux.

Pendant de nombreuses années, le développement d'un système de reconnaissance de formes basé sur le modèle traditionnel a demandé une expertise et des connaissances approfondies pour extraire des données brutes des représentations appropriées qui pourraient être utilisées pour détecter, identifier ou classer des éléments parmi les données d'entrée. Ces méthodes nécessitent des connaissances a priori pour construire un extracteur de caractéristiques adapté aux événements ciblés et à la scène surveillée. Ces contraintes ont conduit à l'émergence de méthodes de détection d'événements anormaux basées sur l'apprentissage de représentations et plus précisément sur l'apprentissage profond. L'apprentissage de représentations ou apprentissage de caractéristiques est un ensemble de techniques permettant d'automatiser l'étape d'extraction de caractéristiques. Ces méthodes permettent de définir, par apprentissage, les transformations appropriées à appliquer aux données d'entrée afin d'obtenir des représentations pour réaliser une tâche ciblée telle que la reconnaissance d'une action, la classification d'une image, l'estimation d'une pose humaine, la segmentation sémantique, etc. Krizhevsky, Sutskever, and Hinton, 2017,Conneau et al., 2016,Szegedy et al., 2015,He et al., 2015.

6.2.1 Transfert des connaissances

Le CNN est un type de réseau neuronal artificiel inspiré du cortex visuel animal. Il est constitué de plusieurs couches qui traitent les données selon un modèle hiérarchique. Il a été démontré qu'un CNN formé pour effectuer une tâche cible peut fournir une fonctionnalité générique et robuste qui peut être utilisée pour effectuer une autre tâche de vision par ordinateur différente de celle pour laquelle il a été spécifiquement entraîné. Dans Sharif Razavian et al., 2014, les représentations extraites avec OverFeat, un CNN entraîné uniquement à la classification d'objets, sont exploitées par un SVM linéaire ou euclidien standard pour différentes tâches (classification de scènes, classification détaillée, détection d'attributs, récupération d'instances visuelles). Les résultats fournissent une preuve tangible de la capacité des CNN à fournir des fonctionnalités génériques et robustes qui peuvent être utilisées pour différentes tâches de vision par ordinateur. Ce principe a été appliqué dans de nombreux travaux sur la détection d'événements anormaux. Dans Bouindour et al., 2017, un CNN 2D préformé à partir de bases de données de classification d'images est modifié pour extraire les représentations de différentes régions des images d'entrée. Un OC-SVM est ensuite utilisé pour détecter lesquelles de ces régions présentent des événements anormaux. Dans Sabokrou et al., 2016, un CNN préformé est combiné à un auto-codeur dispersé qui peut être formé pour fournir un extracteur de caractéristiques à deux niveaux. À la sortie du CNN, un premier classificateur gaussien est utilisé pour classer les régions de l'image comme normales, anormales ou suspectes. Les représentations des régions suspectes sont ensuite transformées par l'auto-codeur pour obtenir des représentations plus discriminantes. Les méthodes basées sur l'apprentissage par transfert ne nécessitent pas de base de données étiquetée pour l'extraction de caractéristiques et leurs résultats en termes de détection et de localisation sont très prometteurs. Cependant, la dépendance de ces méthodes vis-à-vis de modèles pré-entraînés impose

une certaine rigidité qui réduit considérablement leurs perspectives d'améliorations potentielles. Cet inconvénient est à l'origine de l'émergence d'approches basées sur des modèles génératifs et profonds à une classe.

6.2.2 Modèles génératifs

Ces dernières années, l'utilisation des réseaux adversariaux génératifs (GAN) dans l'apprentissage automatique a considérablement augmenté. Le GAN est un algorithme d'apprentissage non supervisé proposé pour la première fois par Goodfellow et al., 2014. Il se compose de deux sous-réseaux, un générateur et un discriminateur concurrent. Pendant la phase d'apprentissage, le générateur essaie de générer des données convaincantes pour tromper le discriminateur qui, à son tour, essaie de détecter si les échantillons générés sont réels (réguliers) ou faux (irréguliers). Dans Lee, Kim, and Ro, 2018, STAN (spatio-temporal adversarial networks) est proposé pour relever le défi de la détection d'anomalies vidéo. Il est composé de deux sous-réseaux, un générateur composé de couches de convolution, ConvLSTM Shi et al., 2015 et de couches de déconvolution et un discriminateur composé de couches de convolution 3D. La détection d'événements anormaux peut être effectuée directement par le discriminateur ou le générateur. Cependant, les meilleurs résultats dans Lee, Kim, and Ro, 2018 sont obtenus en combinant les décisions des deux réseaux. L'auteur de Ravanbakhsh et al., 2017 propose également l'utilisation de GANs pour la détection d'événements anormaux. Un seuillage de l'erreur de génération des deux GAN est utilisé afin d'identifier les régions de l'image contenant les événements anormaux. Le premier GAN est entraîné à générer des représentations de flux optique à partir d'images et le second GAN est entraîné à générer des images à partir de représentations de flux optique. Cependant, l'erreur entre les images générées et les images réelles n'est pas suffisante pour obtenir des résultats convaincants.

6.2.3 Modèles à classe unique

Les approches de détection d'événements anormaux basées sur des modèles reconstructifs, prédictifs ou génératifs reposent généralement sur l'hypothèse qu'un modèle formé sur des images normales ne sera pas capable de reconstruire, de prédire ou de générer des images anormales. Par conséquent, un seuil d'erreur de reconstruction, de prédiction ou de dégénérescence est souvent utilisé pour détecter les événements anormaux. Cependant, dans le cas d'événements vidéo, les différents éléments des situations normales et anormales sont souvent similaires et ce sont généralement leurs interactions ou le contexte qui définissent le caractère normal ou anormal d'une situation. À cet égard, des travaux récents visant à développer des réseaux à classe unique ont été proposés. Sun, Shao, and He, 2019 propose DOC (Deep One-Class), un réseau de neurones convolutifs qui peut être entraîné de bout en bout, en utilisant uniquement des exemples d'apprentissage d'une seule classe. Le réseau est obtenu en remplaçant la softmax habituellement utilisée dans les CNNs par un OC-SVM. De plus, les auteurs définissent une fonction objective qui permet la formation non seulement de la couche OC-SVM, mais aussi de toutes les couches du réseau qui peuvent être formées. De cette façon, le réseau est optimisé pour extraire des représentations compactes et définir l'hyperplan approprié pour isoler les représentations de données de la classe cible. D'autre part, de nombreux travaux basés sur des réseaux de neurones à une classe ont été proposés pour la détection d'anomalies : Chalapathy, Menon, and Chawla, 2018 Ruff et al., 2018 Perera and Patel, 2019. Ces travaux nécessitent très peu d'adaptation pour être utilisés dans le contexte de la détection d'événements vidéo anormaux. Perera and Patel, 2019 propose l'utilisation de l'apprentissage par transfert pour adapter des réseaux pré-entraînés à la détection d'anomalies. Les auteurs partent du principe que deux aspects importants, la

compacité et la description des caractéristiques extraites, doivent être impérativement pris en compte.La description fournit des caractéristiques descriptives. La description fournit des caractéristiques descriptives. Cependant, la compacité est utilisée pour s'assurer que les images de la même classe sont décrites par des représentations similaires. sont décrites par des représentations similaires, de sorte qu'elles sont Elles sont donc positionnées de manière compacte dans l'espace des caractéristiques. Ces deux aspects peuvent contribuer de manière significative à diminuer la distance intra-classe et à augmenter la distance inter-classe. Pour obtenir ces deux aspects, les auteurs proposent deux réseaux. Après l'apprentissage, les deux réseaux identiques sont capables de fournir des représentations à la fois descriptives et compactes. Ces réseaux peuvent être appliqués avec un classificateur à classe unique pour dissocier les éléments d'une classe cible des éléments aberrants. Cependant, ces méthodes proposent d'utiliser des ensembles de données supplémentaires ou des échantillons de flux optique pour analyser le mouvement, ce qui rend ces méthodes dépendantes des caractéristiques artisanales et de la qualité des ensembles de données supplémentaires. Dans ce travail, nous proposons de construire une architecture capable d'analyser le mouvement à partir d'images brutes sans utiliser de jeux de données supplémentaires.

6.3 Transfert et apprentissage non supervisé pour la détection d'anomalies

Dans cette paragraphe , nous proposons une méthode efficace basée sur l'apprentissage profond et l'extraction de caractéristiques spatio-temporelles artisanales pour la détection d'anomalies à l'aide d'un CNN (réseau neuronal à convolution) et d'un HOOF (histogramme de flux optique) pré-entraînés. Les mouvements anormaux sont sélectionnés par seuillage relatif. Un SVM à une classe est entraîné avec des caractéristiques spatiales pour une classification robuste des formes anormales. De plus, une fonction de décision est appliquée à chaque résultat pour corriger les fausses alarmes et les erreurs de détection. Notre méthode est très performante en termes de vitesse et de précision. Elle permet de détecter les anomalies avec une bonne efficacité et réduit la complexité des calculs par rapport aux méthodes de pointe.

6.3.1 Résultats des expériences

UCSD Peds2 et UMN figure 3.10 sont deux ensembles de données différents pour la détection d'anomalies. UCSD Peds2 consiste en des séquences vidéo d'une allée piétonne bondée. Il contient à la fois des événements normaux et anormaux comme les mouvements de marche des cyclistes, des patineurs, des cyclistes et des petits chariots. Dans les allées, le mouvement des piétons dans une zone inattendue est également considéré comme un événement anormal. Il contient 16 échantillons vidéo de formation et 12 échantillons vidéo de test et fournit une vérité de base au niveau de l'image qui nous aide à évaluer la performance de détection en comparant notre méthode avec d'autres méthodes de détection d'anomalies de pointe. D'autre part, le jeu de données UMN est composé de 3 scènes : pelouse (1450 images), intérieur (4415 images) et place (2145 images). Il comporte deux événements : des personnes marchant qui sont considérées comme un événement normal et des personnes courant qui sont considérées comme un événement anormal. La vérité terrain est fournie dans les images vidéo qui doivent être extraites pour évaluer les performances.

Afin d'évaluer la performance de la méthode proposée proposée, nous avons utilisé la mesure de l'ERR (Error), calculée avec l'équation suivante :

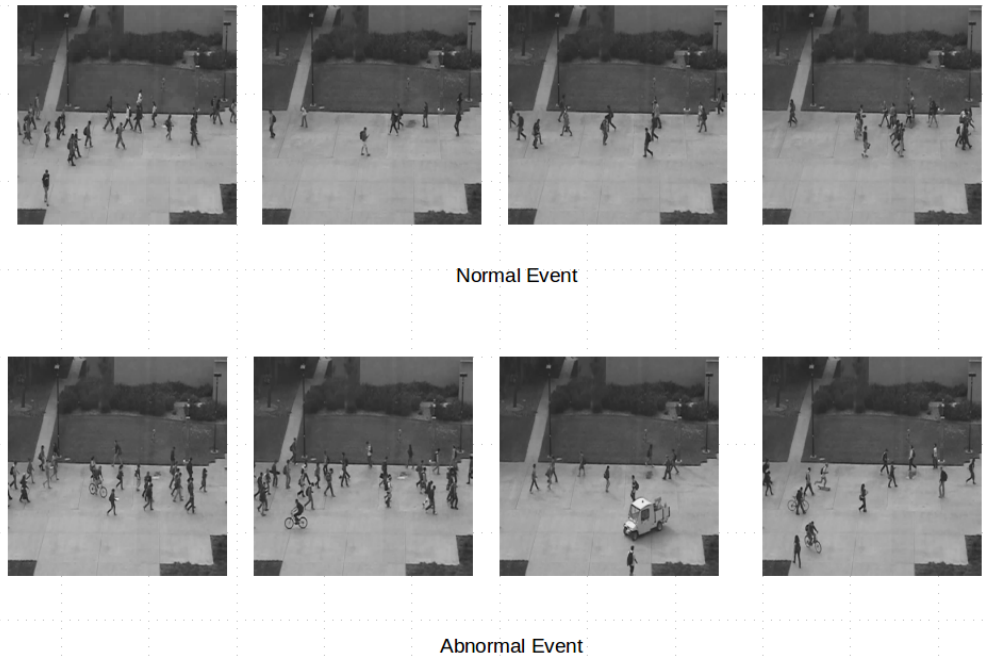


FIGURE 6.1: Jeu de données Ped2

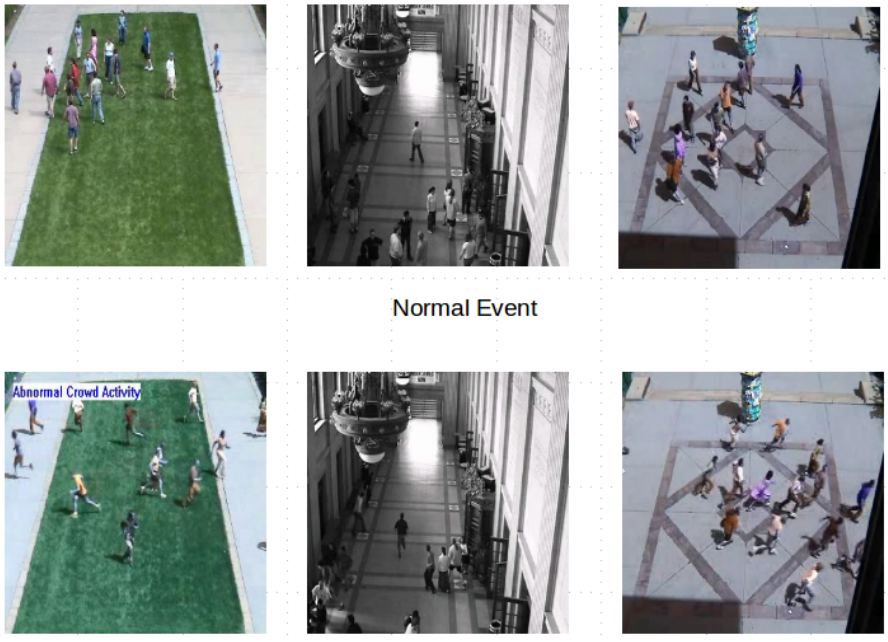


FIGURE 6.2: Jeu de données UMN

$$ERR = \frac{FP + NF}{FN} \tag{6.1}$$

Nos résultats dans chaque dossier de l'USCD Ped 2 sont présentés dans le tableau suivant:

TABLE 6.1: Results in USCD Ped 2 dataset

Folder	nbr frames train	nbr frames test	FP	FN	ERR
1	120	180	12	0	6.6%
2	150	180	95	0	52.7%
3	150	150	3	0	2%
4	180	180	20	0	11.11%
5	180	150	20	0	13.3%
6	150	180	20	4	13.3%
7	150	180	45	0	25%
8	120	180	0	0	0%
9	180	120	0	0	0%
10	180	150	0	2	1.3%
11	180	180	0	0	0%
12	180	180	88	0	48.8%

On peut également noter que pour chaque dossier de test, nous avons utilisé les images du dossier d'entraînement correspondant. Bien que nous ayons entraîné le SVM avec un petit ensemble de données d'environ 150 images pour chaque dossier, notre méthode a obtenu de bons résultats par rapport aux autres méthodes résumées dans le TABLEAU III. Elle a atteint une haute performance et a prouvé son efficacité par rapport à la plupart des méthodes de pointe. Le classificateur SVM à une classe est robuste à la taille de l'ensemble de données d'entraînement, ce qui permet son applicabilité à de très grands ensembles de données vidéo. De plus, ce système hybride diminue les faux négatifs (détections manquées) en dissociant les décisions de tous les descripteurs.

TABLE 6.2: Performance de FC1 et FC2 sur USCD Peds2

Method	ERR
FC1+HOF	16.3%
FC2+HOF	20.19%

Nos résultats en matière de scène de l'UMN sont présentés dans le tableau suivant :

TABLE 6.3: Résultat en UMN

Scene	ERR
Lawn	3.25%
Indoor	3.25%
Plaza	4%

6.4 Réglage fin de CAE pour deep one class

L'apprentissage par transfert dans le contexte des réseaux de neurones nous a permis de développer une méthode efficace pour la détection d'événements anormaux, en prenant en compte les défis des systèmes de vidéosurveillance intelligents. En particulier, nous avons démontré qu'un réseau de neurones pré-entraîné sur de grandes bases de données pour la classification d'actions est particulièrement adapté à la caractérisation des formes et des mouvements dans les séquences vidéo, ce qui en fait des outils efficaces pour la détection d'événements anormaux. Cependant, le fait que cette approche s'appuie sur des réseaux pré-entraînés en mode supervisé sur de grandes bases de données étiquetées peut être préjudiciable. En effet, le fait que le réseau utilisé soit développé et entraîné pour une autre tâche de classification peut induire une inadéquation avec les données cibles. A titre d'exemple, nous pouvons citer les caractéristiques de couleur contenues dans les représentations du réseau qui conduisent à la classification d'images et non utilisées dans nos applications de détection d'événements anormaux. Nous pouvons également supposer que la taille des récepteurs de champ de chaque couche du réseau n'est pas adaptée à la taille des objets présents dans les images de la scène surveillée. D'autre part, l'utilisation de réseaux pré-entraînés impose une certaine inflexibilité à notre approche et réduit considérablement ses perspectives d'amélioration. L'apprentissage non supervisé pourrait être une alternative à l'apprentissage par transfert et pourrait supprimer la dépendance de notre approche vis-à-vis de grandes bases de données étiquetées. L'apprentissage non supervisé est un sous-domaine de l'apprentissage automatique qui, comme son nom l'indique, consiste à apprendre des caractéristiques à partir de données non étiquetées. L'apprentissage non supervisé peut être particulièrement adapté à la détection d'événements vidéo anormaux, car cette tâche est caractérisée par l'indisponibilité de données anormales pendant la phase de formation. Dans cette section, nous allons explorer une stratégie d'exploitation des réseaux de neurones convolutifs en mode non supervisé pour l'extraction de représentations spatio-temporelles, utilisables pour la détection d'anomalies dans des séquences vidéo.

La classification à une classe est un problème d'apprentissage automatique qui a reçu une attention importante de la part de nombreux chercheurs dans différents domaines tels que la détection de nouveauté, la détection d'anomalie et l'imagerie médicale. Néanmoins, le manque de données dans la phase d'apprentissage réduit la profondeur de l'architecture du réseau qui, à son tour, réduit la représentativité des caractéristiques. Pour résoudre cette faiblesse, nous proposons d'affiner un CAE pré-entraîné pour un objectif d'entraînement à une classe, construit à partir du VGG 16 CNN, qui a atteint une précision de 92,7 % dans les cinq premiers tests. La base de données utilisée pour entraîner VGG 16 CNN est ImageNet, un ensemble de données de plus de 14 millions d'images haute résolution appartenant à 1000 classes. Les images ont été collectées sur le Web et étiquetées par des humains à l'aide de l'outil de crowd-sourcing Amazon's Mechanical Turk. Nous gelons les premières couches de convolutions pour exploiter correctement la richesse de la base de données avec laquelle le CNN a été entraîné (Figure 3). L'objectif de l'opération de convolution est d'extraire les caractéristiques de haut niveau de l'image d'entrée. Notre architecture ne doit pas être limitée à une seule couche de convolution. Conventionnellement, la première couche de convolution est chargée de capturer les caractéristiques de bas niveau telles que les bords, la couleur, l'orientation du gradient, etc. En ajoutant des couches, l'architecture s'adapte également aux caractéristiques de haut niveau, ce qui nous donne un réseau qui a une compréhension globale des images dans l'ensemble de données, comme nous le ferions. Ainsi, nous construisons la partie encodeur de notre architecture CAE basée sur les couches de convolution du CNN VGG16 pré-entraîné. Nous gelons le premier bloc de convolution du VGG 16 et nous gardons les autres blocs de convolution entraînables (Figure 3.9). De son côté, la partie décodeur est

un réseau plan composé de quatre couches de déconvolution 2D pour pouvoir reconstruire les images originales, ses hyper-paramètres sont donnés dans le (Tableau 4).

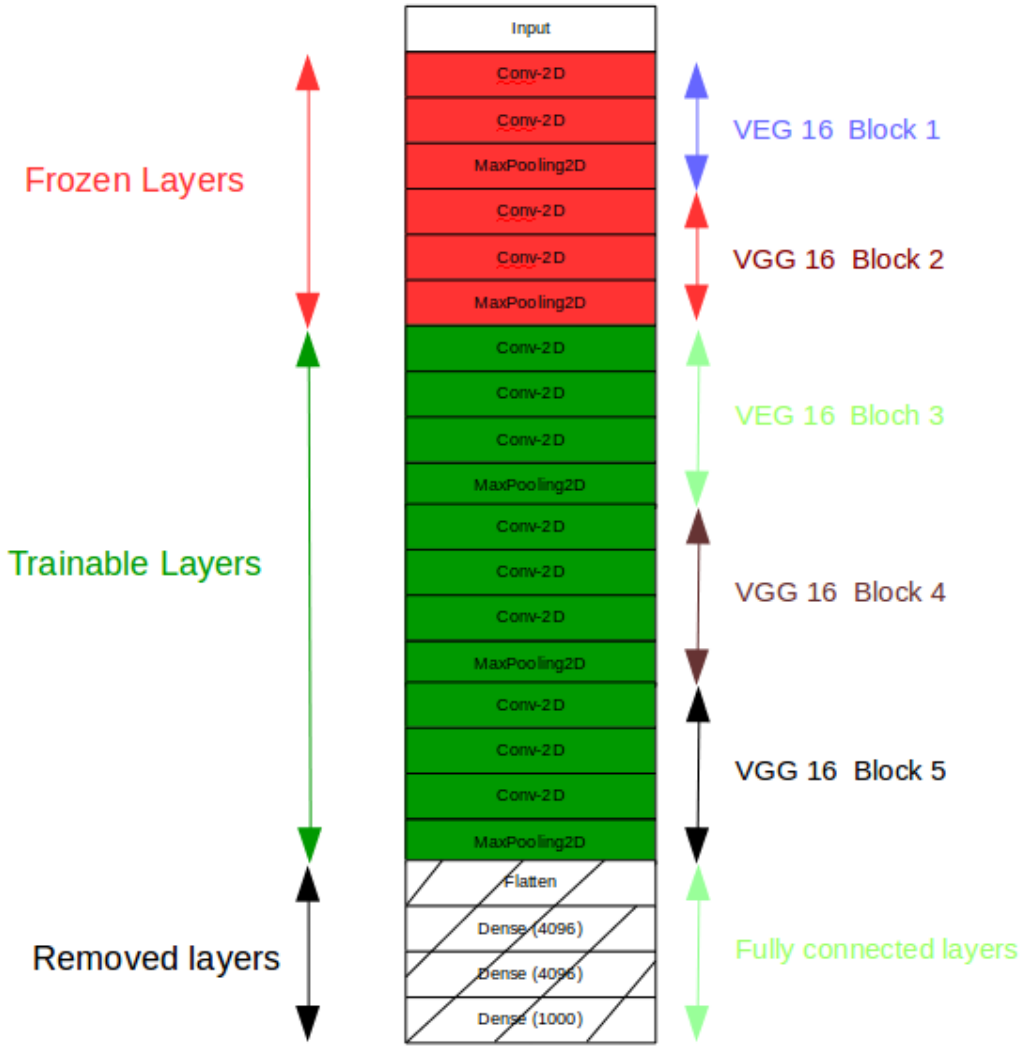


FIGURE 6.3: Architecture VGG 16 utilisée pour le réglage fin de l'objectif d'une classe unique

6.5 Apprentissage profond basé sur le flux optique dans les vidéos de drones

L'architecture que nous proposons se compose de deux parties : un FCN spatio-temporel (ST-FCN) pour apprendre des représentations à partir de trames vidéo, et un FCN de flux optique (OF-FCN) pour renforcer la description du mouvement des représentations apprises. Les deux FCN appris sont obtenus en entraînant deux auto-encodeurs convolutifs (CAE) afin de reconstruire les volumes vidéo et en extrayant la partie encodeur de chacun d'eux.

Le CAE spatio-temporel et le CAE de flux optique sont respectivement appris en utilisant des échantillons d'entraînement normaux et les représentations de flux optique correspondantes. Les deux CAE ont la même architecture et chacun d'eux est composé de quatre couches de convolution 3D (encodeur) et de quatre couches de déconvolution 3D (décodeur). Les couches de convolution encodent les représentations à partir des données d'entrée tandis

TABLE 6.4: Hyper parameter of added layers

Input size	Layer type	Filter Number	Kernel size	Strides	Activation	Output size
[7, 7]	2D-convolution	512	[3,3]	[2,2]	Relu	[3, 3]
[3, 3]	2D-deconvolution	256	[5,5]	[3,3]	Relu	[11,11]
[11,11]	2D-deconvolution	128	[5,5]	[2,2]	Relu	[35,35]
[35,35]	2D-deconvolution	96	[7,7]	[2,2]	Relu	[109,109]
[109, 109]	2D-deconvolution	1	[8,8]	[2,2]	linear	[224, 224]

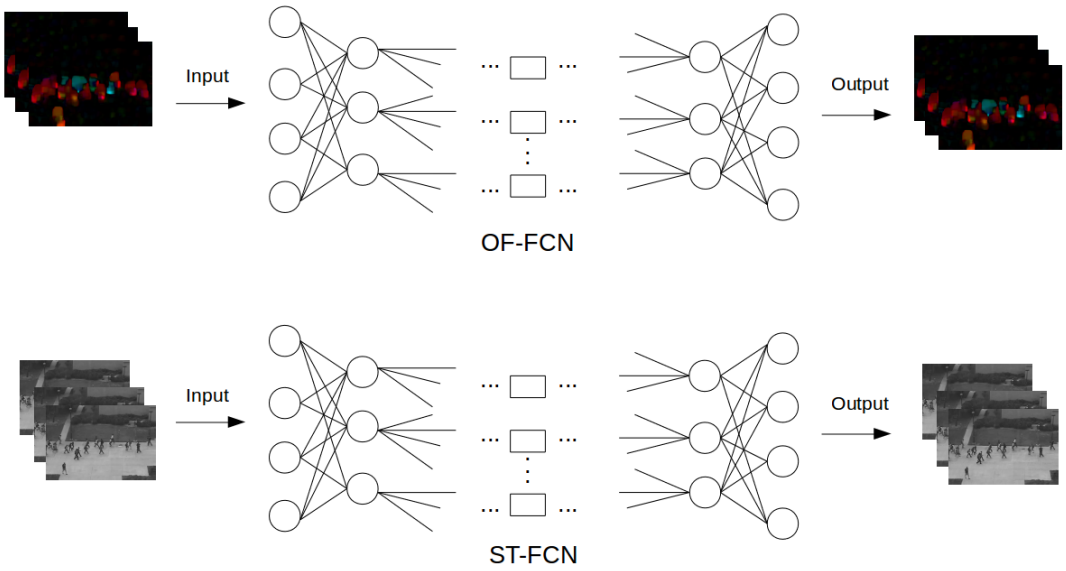


FIGURE 6.4: Architecture TS-FCN 1

que la déconvolution reflète la partie encodeur pour les reconstruire. L'IAO spatio-temporelle prend en entrée les volumes 3D de trois images consécutives F: $\{F_t ; F_{t-1} ; F_{t-2}\}$. L'CAE de flux optique prend en entrée les volumes 3D de trois représentations de flux optique OF: $\{OF_t ; OF_{t-1} ; OF_{t-2}\}$, où OF_t est obtenu en extrayant le flux optique pour chacune des deux images consécutives

L'extraction des images de flux optique dans la phase de test permet au système d'exécuter une tâche supplémentaire pour extraire les images de flux optique. De plus, dans la phase d'entraînement, les représentations des deux volumes de deux flux sont indépendantes. Ensuite, nous proposons une deuxième méthode basée sur une nouvelle architecture d'un bloc pour représenter notre TS-FCN afin de corriger les imperfections de la première méthode. Ce TS-FCN est appris en utilisant des représentations d'échantillons d'entraînement normaux d'images originales uniquement. Il est composé de huit couches de convolution 3D (encodeur), huit couches de déconvolution 3D (décodeur) et une couche de concaténation pour combiner les deux présentations. Le TS-FCN prend en entrée les volumes 3D de trois images consécutives F: $\{F_t ; F_{t-1} ; F_{t-2}\}$ et essaie non seulement de reconstruire ces images mais aussi de reconstruire les volumes 3D du flux optique OF: $\{OF_t ; OF_{t-1} ; OF_{t-2}\}$ en même temps. L'erreur quadratique moyenne est utilisée comme fonction de perte pour entraîner notre modèle. Après la phase d'entraînement, la partie encodeur contenant 8 couches de convolutions représente notre TS-FCN. Notre modèle fournit une carte de caractéristiques de dimension 676*512 dans ce cas. Il est capable d'obtenir une représentation spatio-temporelle robuste de chaque forme et mouvement dans les images. Dans la phase de test, nous n'avons

pas besoin d'extraire la représentation du flux optique manuellement mais notre modèle est capable de construire une nouvelle représentation du flux optique à partir des images originales qui sont plus dédiées à la tâche de détection des anomalies.

TABLE 6.5: EER and AUC dans la base de données Ped2

Methods	EER	AUC
PCA	29.20	73.98
CAE(FR)	26.00	81.4
ConvAE	21.7	90.00
EAD	16.47	86.43
Chong	12	-
Sabokrou	8.2	-
ours		
ST-FCN	19	87.15
TS-FCN 1	**13.2**	**91.6**
TS-FCN 2	**8.45**	**93.6**

Ces dernières années, l'utilisation de drones pour des tâches de surveillance a augmenté dans le monde entier. Cependant, dans le contexte de la détection d'anomalies, seuls les événements normaux sont disponibles pour le processus d'apprentissage. Par conséquent, la mise en œuvre d'une méthode d'apprentissage générative en mode non supervisé pour résoudre ce problème devient fondamentale. Dans ce contexte, nous proposons une nouvelle architecture de bout en bout capable de générer des images de flux optique à partir d'images originales de drones et d'extraire des caractéristiques spatio-temporelles compactes à fin de détecter les anomalies. Elle est conçue avec une fonction de perte personnalisée comme une somme de trois termes, la perte de reconstruction (R_l), la perte de génération (G_l) et la perte de compacité (C_l) pour assurer une classification efficace de la classe "deep-one". De plus, nous proposons de minimiser l'effet du mouvement des drones dans le traitement vidéo en appliquant une soustraction du fond sur les images de flux optique. Nous avons testé notre méthode sur des jeux de données très complexes appelés jeux de données vidéo de mini-drones et avons obtenu des résultats surpassant les performances des techniques existantes avec une AUC de 85,3.

Dans cette section, nous proposons une nouvelle architecture non supervisée de bout en bout pour la détection d'anomalies dans les séquences vidéo de drones. Elle est entraînée avec seulement des images consécutives normales et de flux optique. Notre architecture est capable de construire de nouvelles représentations de flux optique d'une vidéo de drone à partir de trames originales consécutives. Elle est basée sur un mélange de couches de convolution et de déconvolution capables non seulement de générer automatiquement des images de flux optique mais aussi d'extraire des caractéristiques compactes des images originales et des images de flux optique pendant la phase de test. Le calcul classique du flux optique est alors évité et remplacé par un réseau neuronal rapide et efficace basé sur la convolution/déconvolution. La procédure proposée peut produire des représentations de flux optique d'échantillons anormaux avec une génération d'erreur de flux optique (OFE) plus élevée que celle des échantillons normaux, intuitivement en diminuant la distance intra-classe de la classe normale pendant la phase de formation, comme dans l'équation suivante :

$$OFE = \frac{1}{n}\sum_{1}^{n}(\phi(i) - \hat{\phi}(i))^2 \tag{6.2}$$

Où $\phi(i)$ est le flux optique original et $\hat{\phi}(i)$ est le flux optique généré. Grâce à cette architecture, notre modèle est capable de représenter correctement les formes et le mouvement dans les vidéos. Le réseau neuronal est composé de 8 couches de convolution, d'une couche de concaténation, pour combiner les cartes de caractéristiques de chacune des 4 couches de convolution, et de 8 couches de déconvolution pour reconstruire l'entrée composée des images originales consécutives et pour générer les images de flux optiques consécutives. La couche de concaténation est notre couche goulot d'étranglement. Nous avons appelé notre architecture un générateur de flux optique CNN en raison de sa capacité à générer des échantillons de flux optique à partir d'images originales.

TABLE 6.6: Our architecture hyperparameters

Layer	Filters	Kernel (h,w,d)	Stride(h,w,d)
Conv1	64	[11,11,1]	[2,2,1]
Conv2	128	[3,3,1]	[1,1,1]
Conv3	256	[3,3,3]	[2,2,1]
Conv4	512	[3,3,1]	[2,2,1]
Conv5	64	[11,11,1]	[2,2,1]
Conv6	128	[3,3,1]	[1,1,1]
Conv7	256	[3,3,3]	[2,2,1]
Conv8	512	[3,3,1]	[2,2,1]
Concat	1024	—-	——
Deconv1	512	[3,3,1]	[2,2,1]
Deconv2	256	[3,3,3]	[2,2,1]
Deconv3	128	[3,3,1]	[1,1,1]
Deconv4	1	[11,11,1]	[2,2,1]
Deconv5	512	[3,3,1]	[2,2,1]
Deconv6	256	[3,3,3]	[2,2,1]
Deconv7	128	[3,3,1]	[1,1,1]
Deconv8	1	[11,11,1]	[2,2,1]

6.6 Conclusion

La majorité de nos travaux présentés dans cette thèse sont basés sur les réseaux de neurones convolutifs. Cependant, la conception d'un réseau de neurones n'est pas une tâche facile. De nombreux choix affectent les performances du réseau. Ceux-ci incluent la manière d'échantillonner et de prétraiter les données d'entrée, le nombre de couches, leurs types et les différents paramètres à leur appliquer, l'optimiseur à utiliser pour l'entraînement du réseau et ses paramètres, la longueur de la séquence temporelle à utiliser...ect. Outre le nombre de paramètres, l'apprentissage d'un réseau est non seulement coûteux en termes de ressources matérielles (GPU), mais aussi en temps. Les différents paramètres d'un réseau sont étroitement liés aux données d'apprentissage, ce qui signifie que pour différents ensembles de données, ces paramètres sont très différents. Cet aspect, ajouté aux difficultés de conception et d'entraînement des réseaux neuronaux, nous a contraints à limiter les jeux de données de test pour nos différentes méthodes. Dans nos travaux futurs, nous prévoyons d'explorer d'autres ensembles de données pour confirmer la précision et la pertinence de nos approches. Le fait de n'utiliser que des échantillons normaux lors de la formation peut limiter les performances de la classification profonde à une classe, car il n'y a pas d'échantillons anormaux utilisés pour augmenter l'inter-distance entre les caractéristiques normales et anormales. Pour cette raison, nous avons besoin de plus d'échantillons des deux classes. Dans nos futurs travaux,

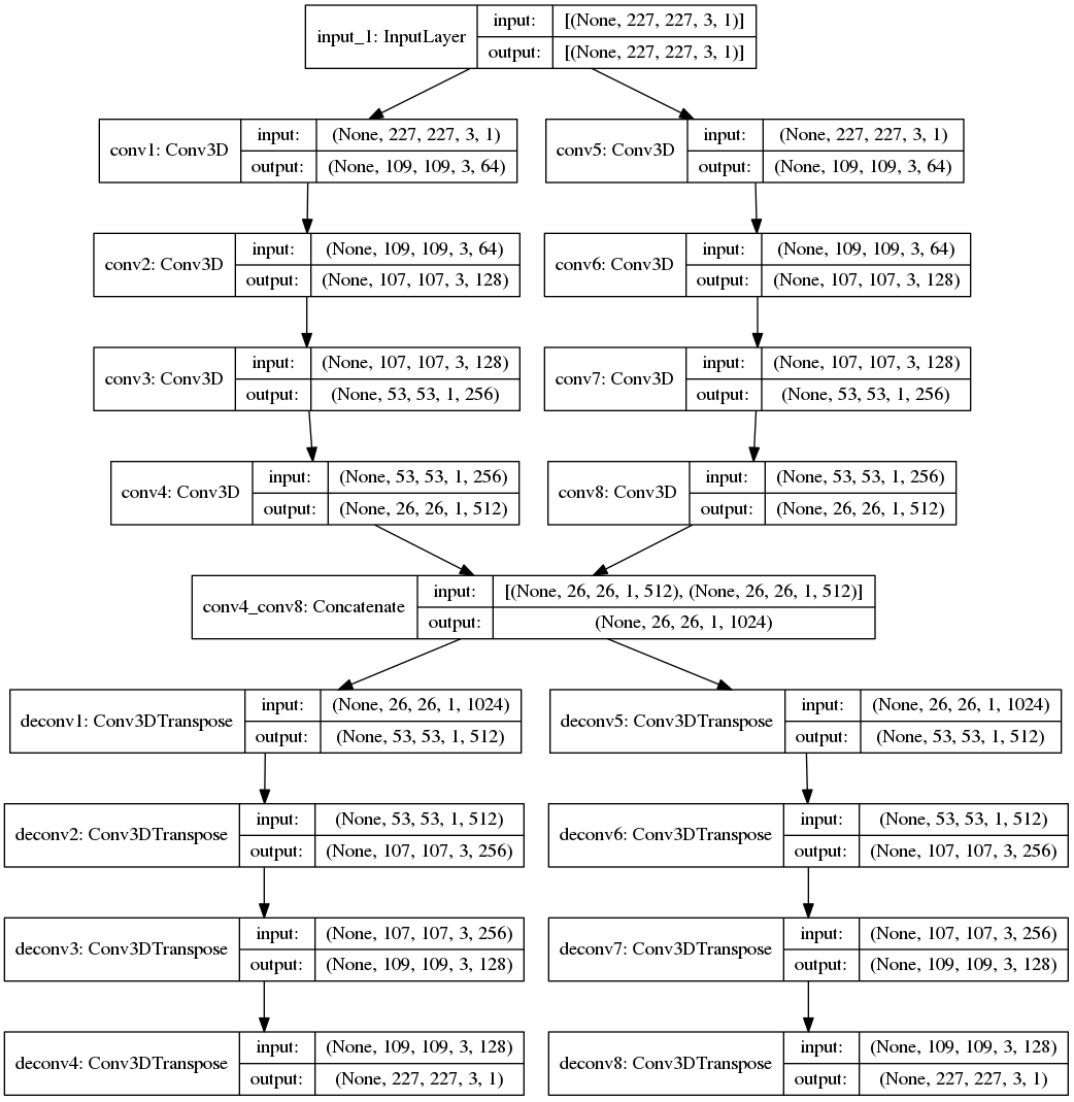


FIGURE 6.5: Notre architecture

nous étudierons le Deep Fake, par exemple, l'apprentissage par transfert d'art pour générer des échantillons anormaux à partir d'événements normaux. Dans cet thèse, nous proposons une nouvelle méthode d'apprentissage non supervisé basée sur une architecture profonde de bout en bout pour la détection d'anomalies dans les flux vidéo de drones. Le principal avantage de cette méthode est son efficacité à extraire conjointement les caractéristiques du flux optique et à intégrer un terme de régularisation de la compacité pendant l'apprentissage. Cette méthode s'avère prometteuse en termes de détection et de localisation d'anomalies par les caméras de drones et donne des résultats expérimentaux très performants par rapport aux méthodes de l'état de l'art. Notre travail futur est d'étudier ces résultats en mettant en place un ordinateur embarqué sur le drone pour une application de détection d'anomalies en temps réel.

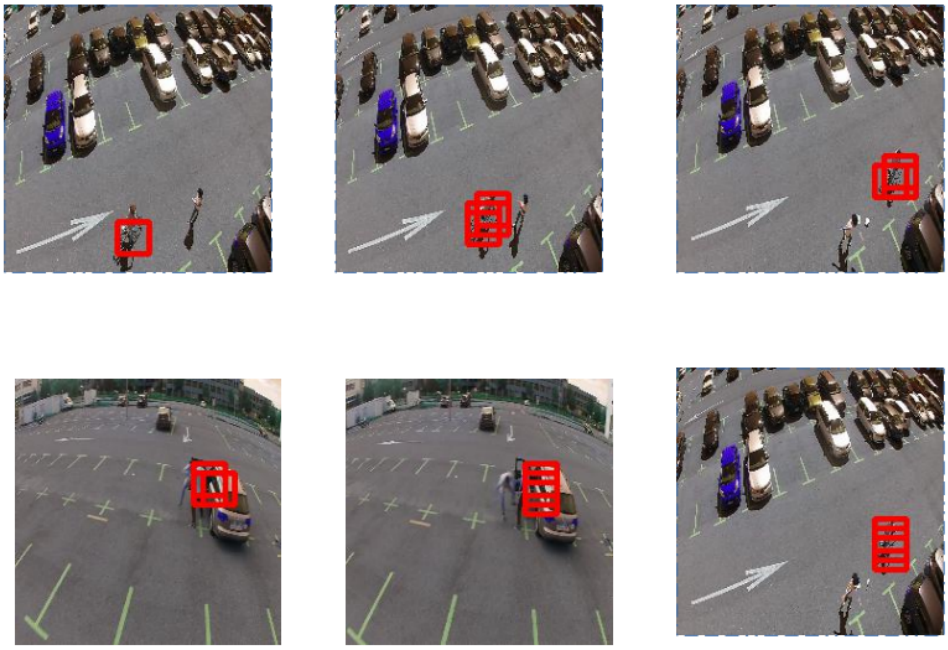


FIGURE 6.6: Nos résultat sur MDVD

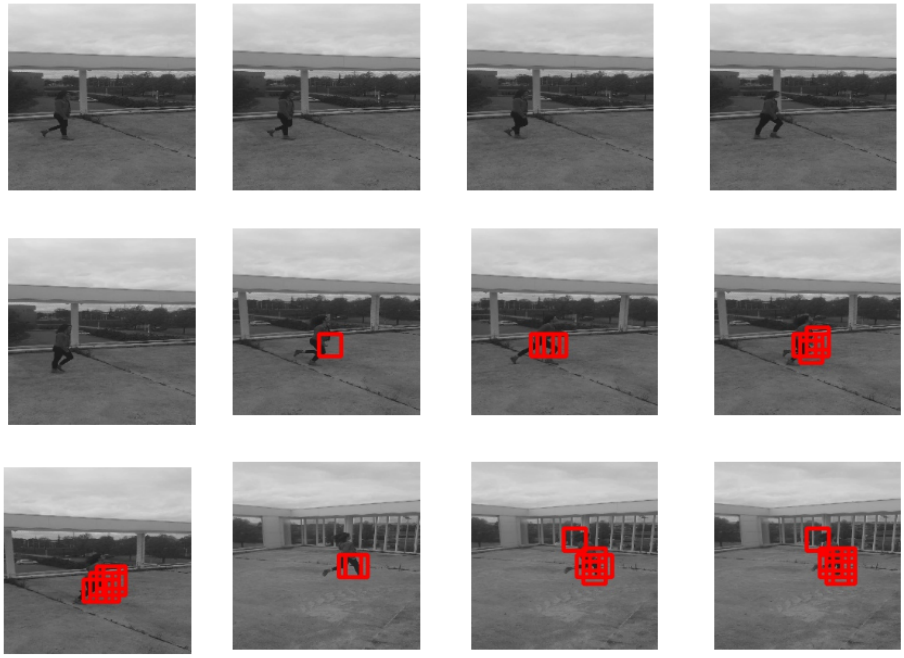


FIGURE 6.7: Nos résultat sur running dataset

Bibliography

Adam, A. et al. (2008). "Robust Real-Time Unusual Event Detection using Multiple Fixed-Location Monitors". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30.3, pp. 555–560. DOI: 10.1109/TPAMI.2007.70825.

Adam, Amit et al. (2008). "Robust real-time unusual event detection using multiple fixed-location monitors". In: *IEEE transactions on pattern analysis and machine intelligence* 30.3, pp. 555–560.

Aggarwal, Jake K. (2011). "Recognition of Human Activities". In: *Combinatorial Image Analysis*. Ed. by Jake K. Aggarwal et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 1–4. ISBN: 978-3-642-21073-0.

Amodei, Dario et al. (2015). "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin". In: *CoRR* abs/1512.02595. arXiv: 1512.02595. URL: http://arxiv.org/abs/1512.02595.

Andrews, Jerone et al. (2016). "Transfer representation-learning for anomaly detection". In: JMLR.

Barenji, Ali Vatankhah et al. (2019). "Intelligent E-commerce logistics platform using hybrid agent based approach". In: *Transportation Research Part E: Logistics and Transportation Review* 126, pp. 15 –31. ISSN: 1366-5545. DOI: https://doi.org/10.1016/j.tre.2019.04.002. URL: http://www.sciencedirect.com/science/article/pii/S1366554518305349.

Beltramelli, Tony (2017). "pix2code: Generating Code from a Graphical User Interface Screenshot". In: *CoRR* abs/1705.07962. arXiv: 1705.07962. URL: http://arxiv.org/abs/1705.07962.

Benezeth, Yannick, Pierre-Marc Jodoin, and Venkatesh Saligrama (2011). "Abnormality detection using low-level co-occurring events". In: *Pattern Recognition Letters* 32.3, pp. 423 –431. ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2010.10.008. URL: http://www.sciencedirect.com/science/article/pii/S0167865510003545.

Bengio, Yoshua (2012). "Deep learning of representations for unsupervised and transfer learning". In: *Proceedings of ICML workshop on unsupervised and transfer learning*. JMLR Workshop and Conference Proceedings, pp. 17–36.

Bertini, Marco, Alberto Del Bimbo, and Lorenzo Seidenari (2012). "Multi-scale and real-time non-parametric approach for anomaly detection and localization". In: *Computer Vision and Image Understanding* 116.3, pp. 320–329.

Bilen, H. et al. (2016). "Dynamic Image Networks for Action Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3034–3042. DOI: 10.1109/CVPR.2016.331.

Biliotti, D., G. Antonini, and J. P. Thiran (2005). "Multi-Layer Hierarchical Clustering of Pedestrian Trajectories for Automatic Counting of People in Video Sequences". In: *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05) - Volume 1*. Vol. 2, pp. 50–57. DOI: 10.1109/ACVMOT.2005.77.

Bobick, A. F. and J. W. Davis (2001). "The recognition of human movement using temporal templates". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.3, pp. 257–267. DOI: 10.1109/34.910878.

Boiman, Oren and Michal Irani (2007). "Detecting irregularities in images and in video". In: *International journal of computer vision* 74.1, pp. 17–31.

Bonetto, M. et al. (2015). "Privacy in mini-drone based video surveillance". In: *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. Vol. 04, pp. 1–6. DOI: 10.1109/FG.2015.7285023.

Boser, Bernhard E., Isabelle M. Guyon, and Vladimir N. Vapnik (1992). "A Training Algorithm for Optimal Margin Classifiers". In: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. COLT '92. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 144–152. ISBN: 089791497X. DOI: 10.1145/130385.130401. URL: https://doi.org/10.1145/130385.130401.

Bouindour, S. et al. (2017). "Abnormal event detection using convolutional neural networks and 1-class SVM classifier". In: *8th International Conference on Imaging for Crime Detection and Prevention (ICDP 2017)*, pp. 1–6. DOI: 10.1049/ic.2017.0040.

Bouindour, Samir et al. (2017). "Abnormal event detection using convolutional neural networks and 1-class SVM classifier". In:

Bouindour S, Snoussi H Hittawe M.M Tazi N Wang-T. (2019). "An On-Line and Adaptive Method for Detecting Abnormal Events in Videos Using Spatio-Temporal ConvNet". In: *Appl. Sci. 2019, 9, 757*.

Calderara, Simone et al. (2011). "Detecting anomalies in people's trajectories using spectral graph analysis". In: *Computer Vision and Image Understanding* 115.8, pp. 1099 –1111. ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2011.03.003. URL: http://www.sciencedirect.com/science/article/pii/S1077314211000919.

Canny, J. (1986). "A Computational Approach to Edge Detection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8.6, pp. 679–698. DOI: 10.1109/TPAMI.1986.4767851.

Chalapathy, Raghavendra, Aditya Krishna Menon, and Sanjay Chawla (2018). "Anomaly detection using one-class neural networks". In: *arXiv preprint arXiv:1802.06360*.

Cheng, K., Y. Chen, and W. Fang (2015). "Video anomaly detection and localization using hierarchical feature representation and Gaussian process regression". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2909–2917. DOI: 10.1109/CVPR.2015.7298909.

Chong, Y. S. and Y. H. Tay (2017a). "Abnormal event detection in videos using spatiotemporal autoencoder". In: inInternational Symposium on Neural Networks, pp. 189–196, Springer.

Chong, Yong Shean and Yong Haur Tay (2017c). "Abnormal event detection in videos using spatiotemporal autoencoder". In: *International Symposium on Neural Networks*. Springer, pp. 189–196.

— (2017b). "Abnormal Event Detection in Videos using Spatiotemporal Autoencoder". In: *CoRR* abs/1701.01546. arXiv: 1701.01546. URL: http://arxiv.org/abs/1701.01546.

Chung, Joon Son et al. (2016). "Lip Reading Sentences in the Wild". In: *CoRR* abs/1611.05358. arXiv: 1611.05358. URL: http://arxiv.org/abs/1611.05358.

Chung, Pau-Choo and Chin-De Liu (2008). "A daily behavior enabled hidden Markov model for human behavior understanding". In: *Pattern Recognition* 41.5, pp. 1572 –1580. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2007.10.022. URL: http://www.sciencedirect.com/science/article/pii/S0031320307004700.

Cong, Yang, Junsong Yuan, and Ji Liu (2011). "Sparse reconstruction cost for abnormal event detection". In: *CVPR 2011*. IEEE, pp. 3449–3456.

Conneau, Alexis et al. (2016). "Very Deep Convolutional Networks for Natural Language Processing". In: *CoRR* abs/1606.01781. arXiv: 1606.01781. URL: http://arxiv.org/abs/1606.01781.

D.-S. Pham, B. Saha D. Q. Phung S. Venkatesh (2011). "Detection of Cross-channel Anomalies from Multiple Data Channels". In: nICDM,2011, pp. 527–536.

Dalal, Navneet (2006). "Finding people in images and videos". PhD thesis. Institut National Polytechnique de Grenoble-INPG.

Dalal, Navneet, Bill Triggs, and Cordelia Schmid (2006). "Human detection using oriented histograms of flow and appearance". In: *European conference on computer vision*. Springer, pp. 428–441.

Davis, J. W. (2001). "Hierarchical motion history images for recognizing human motion". In: *Proceedings IEEE Workshop on Detection and Recognition of Events in Video*, pp. 39–46. DOI: 10.1109/EVENT.2001.938864.

Dee H.M., Velastin S.A (2008). "How close are we to solving the problem of automated visual surveillance ?" In: *Machine Vision and Applications* 19, pp. 329–343. DOI: https://doi.org/10.1007/s00138-007-0077-z.

Deng, Jia et al. (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255.

Ding, Chunhui et al. (2014). "Violence Detection in Video by Using 3D Convolutional Neural Networks". In: *Advances in Visual Computing*. Ed. by George Bebis et al. Cham: Springer International Publishing, pp. 551–558. ISBN: 978-3-319-14364-4.

Donahue, Jeff et al. (2014). "Decaf: A deep convolutional activation feature for generic visual recognition". In: *International conference on machine learning*. PMLR, pp. 647–655.

Dong, N. et al. (2010). "Traffic Abnormality Detection through Directional Motion Behavior Map". In: *2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 80–84. DOI: 10.1109/AVSS.2010.61.

Duque, D., H. Santos, and P. Cortez (2007). "Prediction of Abnormal Behaviors for Intelligent Video Surveillance Systems". In: *2007 IEEE Symposium on Computational Intelligence and Data Mining*, pp. 362–367. DOI: 10.1109/CIDM.2007.368897.

Ehsani, Kiana, Roozbeh Mottaghi, and Ali Farhadi (2017). "SeGAN: Segmenting and Generating the Invisible". In: *CoRR* abs/1703.10239. arXiv: 1703.10239. URL: http://arxiv.org/abs/1703.10239.

Farnebäck, Gunnar (2003). "Two-frame motion estimation based on polynomial expansion". In: *Scandinavian conference on Image analysis*. Springer, pp. 363–370.

Feng, J., C. Zhang, and P. Hao (2010). "Online Learning with Self-Organizing Maps for Anomaly Detection in Crowd Scenes". In: *2010 20th International Conference on Pattern Recognition*, pp. 3599–3602. DOI: 10.1109/ICPR.2010.878.

Foroughi, H., A. Rezvanian, and A. Paziraee (2008). "Robust Fall Detection Using Human Shape and Multi-class Support Vector Machine". In: *2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing*, pp. 413–420. DOI: 10.1109/ICVGIP.2008.49.

Gong, Y. et al. (2013). "Iterative Quantization: A Procrustean Approach to Learning Binary Codes for Large-Scale Image Retrieval". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.12, pp. 2916–2929. DOI: 10.1109/TPAMI.2012.193.

Goodfellow, Ian J. et al. (2014). *Generative Adversarial Networks*. arXiv: 1406.2661 [stat.ML].

Gutoski, M. et al. (2017). "Detection of Video Anomalies Using Convolutional Autoencoders and One-Class Support Vector Machines". In: *Anais do 13 Congresso Brasileiro de Inteligência Computacional*. Ed. by L. Martí and N. Sánchez Pi. Curitiba, PR: ABRICOM, pp. 1–12.

Hamdi, Slim et al. (2019). "Hybrid deep learning and HOF for Anomaly Detection". In: *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, pp. 575–580.

Hasan, Mahmudul et al. (2016). "Learning Temporal Regularity in Video Sequences". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

He, Kaiming et al. (2015). "Deep Residual Learning for Image Recognition". In: *CoRR* abs/1512.03385. arXiv: 1512.03385. URL: http://arxiv.org/abs/1512.03385.

Henrio, Jordan and Tomoharu Nakashima (2018). "Anomaly Detection in Videos Recorded by Drones in a Surveillance Context". In: *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, pp. 2503–2508.

Hung Vu, Tu Dinh Nguyen Dinh Phung (2018). "Detection of Unknown Anomalies in Streaming Videos with Generative Energy-based Boltzmann Models". In: arXiv preprint arXiv:1805.01090.

Isola, Phillip et al. (2016). "Image-to-Image Translation with Conditional Adversarial Networks". In: *CoRR* abs/1611.07004. arXiv: 1611.07004. URL: http://arxiv.org/abs/1611.07004.

Jamadandi, Adarsh, Sunidhi Kottursheттar, and Uma Mudenagudi (2020). "Two Stream Convolutional Neural Networks for Anomaly Detection in Surveillance Videos". In: *Smart Computing Paradigms: New Progresses and Challenges*. Ed. by Atilla Elçi et al. Singapore: Springer Singapore, pp. 41–48. ISBN: 978-981-13-9683-0.

Javan Roshtkhari, Mehrsan and Martin D. Levine (2013). "An on-line, real-time learning method for detecting anomalies in videos using spatio-temporal compositions". In: *Computer Vision and Image Understanding* 117.10, pp. 1436 –1452. ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2013.06.007. URL: http://www.sciencedirect.com/science/article/pii/S1077314213001239.

Jiang, Fan et al. (2011). "Anomalous video event detection using spatiotemporal context". In: *Computer Vision and Image Understanding* 115.3. Special issue on Feature-Oriented Image and Video Computing for Extracting Contexts and Semantics, pp. 323 –333. ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2010.10.008. URL: http://www.sciencedirect.com/science/article/pii/S1077314210002390.

Jin, Yanghua et al. (2017). "Towards the Automatic Anime Characters Creation with Generative Adversarial Networks". In: *CoRR* abs/1708.05509. arXiv: 1708.05509. URL: http://arxiv.org/abs/1708.05509.

Karpathy, Andrej et al. (2014). "Large-scale video classification with convolutional neural networks". In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1725–1732.

Keval, H. and M. Sasse (2006). "Man or a Gorilla? Performance Issues with CCTV Technology in Security Control Rooms". In:

Kim, J. and K. Grauman (2009). "Observe locally, infer globally: A space-time MRF for detecting abnormal activities with incremental updates". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2921–2928. DOI: 10.1109/CVPR.2009.5206569.

Kim, Jaechul and Kristen Grauman (2009a). "Observe locally, infer globally: a space-time MRF for detecting abnormal activities with incremental updates". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2921–2928.

— (2009b). "Observe locally, infer globally: a space-time MRF for detecting abnormal activities with incremental updates". In: *2009 IEEE conference on computer vision and pattern recognition*. IEEE, pp. 2921–2928.

Kiran, B. Ravi, Dilip Mathew Thomas, and Ranjith Parakkal (2018). "An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos".

In: *CoRR* abs/1801.03149. arXiv: 1801.03149. URL: http://arxiv.org/abs/1801.03149.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton (May 2017). "ImageNet Classification with Deep Convolutional Neural Networks". In: *Commun. ACM* 60.6, 84–90. ISSN: 0001-0782. DOI: 10.1145/3065386. URL: https://doi.org/10.1145/3065386.

Kwak, Sooyeong and Hyeran Byun (2011). "Detection of dominant flow and abnormal events in surveillance video". In: *Optical Engineering* 50.2, p. 027202.

Lao, W., J. Han, and P. H. n. De With (2009). "Automatic video-based human motion analyzer for consumer surveillance system". In: *IEEE Transactions on Consumer Electronics* 55.2, pp. 591–598. DOI: 10.1109/TCE.2009.5174427.

Laptev, Ivan et al. (2008). "Learning realistic human actions from movies". In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1–8.

LeCun Y., Bengio Y. Hinton G. (2015). "Deep learning". In: *Nature 521, 436–444*.

LeCun, Yann (2016). "L'apprentissage profond, une révolution en intelligence artificielle". In: *La lettre du Collège de France [En ligne]* 41. URL: http://journals.openedition.org/lettre-cdf/3227.

LeCun, Yann et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.

Ledig, Christian et al. (2016). "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network". In: *CoRR* abs/1609.04802. arXiv: 1609.04802. URL: http://arxiv.org/abs/1609.04802.

Lee, Sang-gil et al. (2017). "A SeqGAN for Polyphonic Music Generation". In: *CoRR* abs/1710.11418. arXiv: 1710.11418. URL: http://arxiv.org/abs/1710.11418.

Lee, Sangmin, Hak Gu Kim, and Yong Man Ro (2018). "STAN: Spatio-Temporal Adversarial Networks for Abnormal Event Detection". In: *CoRR* abs/1804.08381. arXiv: 1804.08381. URL: http://arxiv.org/abs/1804.08381.

Li, Jianan et al. (2017). "Perceptual Generative Adversarial Networks for Small Object Detection". In: *CoRR* abs/1706.05274. arXiv: 1706.05274. URL: http://arxiv.org/abs/1706.05274.

Li, Nannan et al. (2015). "Spatio-temporal context analysis within video volumes for anomalous-event detection and localization". In: *Neurocomputing* 155, pp. 309 –319. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2014.12.064. URL: http://www.sciencedirect.com/science/article/pii/S0925231214017287.

Li, Weixin, Vijay Mahadevan, and Nuno Vasconcelos (2013). "Anomaly detection and localization in crowded scenes". In: *IEEE transactions on pattern analysis and machine intelligence* 36.1, pp. 18–32.

Liu, Wen et al. (2018). "Future frame prediction for anomaly detection–a new baseline". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6536–6545.

Liu, Xingchen et al. (2019). "Fault diagnosis of rotating machinery under noisy environment conditions based on a 1-D convolutional autoencoder and 1-D convolutional neural network". In: *Sensors* 19.4, p. 972.

Long, Jonathan, Evan Shelhamer, and Trevor Darrell (2015). "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440.

Lucas, Bruce and Takeo Kanade (Apr. 1981). "An Iterative Image Registration Technique with an Application to Stereo Vision (IJCAI)". In: vol. 81.

M. Hasan, J. Choi J. Neumann A. K. Roy-Chowdhury L. S. Davis (2016). ""Learning Temporal Regularity in Video Sequences," in: in CVPR, 2016.

M. Ribeiro, A. E. Lazzaretti H. S. Lopes (2017). "A Study of Deep Convolutional Auto-Encoders for Anomaly Detection in Videos". In: Pattern Recognition Letters.

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

Mahadevan, V. et al. (2010). "Anomaly detection in crowded scenes". In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1975–1981. DOI: 10.1109/CVPR.2010.5539872.

Masci, Jonathan et al. (2011). "Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction". In: *Artificial Neural Networks and Machine Learning – ICANN 2011*. Ed. by Timo Honkela et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 52–59. ISBN: 978-3-642-21735-7.

Medel, Jefferson Ryan and Andreas E. Savakis (2016). "Anomaly Detection in Video Using Predictive Convolutional Long Short-Term Memory Networks". In: *CoRR* abs/1612.00390. arXiv: 1612.00390. URL: http://arxiv.org/abs/1612.00390.

Mehran, Ramin, Alexis Oyama, and Mubarak Shah (2009a). "Abnormal crowd behavior detection using social force model". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 935–942.

— (2009b). "Abnormal crowd behavior detection using social force model". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 935–942.

Nanni, Loris, Stefano Ghidoni, and Sheryl Brahnam (2017). "Handcrafted vs. non-handcrafted features for computer vision classification". In: *Pattern Recognition* 71, pp. 158–172.

Oza, P. and V. M. Patel (2019a). "One-Class Convolutional Neural Network". In: *IEEE Signal Processing Letters* 26.2, pp. 277–281. DOI: 10.1109/LSP.2018.2889273.

— (2019b). "One-Class Convolutional Neural Network". In: *IEEE Signal Processing Letters* 26.2, pp. 277–281.

Patil, N. and P. K. Biswas (2016). "A survey of video datasets for anomaly detection in automated surveillance". In: *2016 Sixth International Symposium on Embedded Computing and System Design (ISED)*, pp. 43–48. DOI: 10.1109/ISED.2016.7977052.

Perera, P. and V. M. Patel (2019). "Learning Deep Features for One-Class Classification". In: *IEEE Transactions on Image Processing* 28.11, pp. 5450–5463.

Perera, Pramuditha and Vishal M Patel (2019). "Learning deep features for one-class classification". In: *IEEE Transactions on Image Processing* 28.11, pp. 5450–5463.

Pham, Duc Son et al. (2011). "Detection of cross-channel anomalies from multiple data channels". In: *2011 IEEE 11th International Conference on Data Mining*. IEEE, pp. 527–536.

Piciarelli, C., G. L. Foresti, and L. Snidaro (2005). "Trajectory clustering and its applications for video surveillance". In: *IEEE Conference on Advanced Video and Signal Based Surveillance, 2005.* Pp. 40–45. DOI: 10.1109/AVSS.2005.1577240.

Piciarelli, C. and G.L. Foresti (2006). "On-line trajectory clustering for anomalous events detection". In: *Pattern Recognition Letters* 27.15. Vision for Crime Detection and Prevention, pp. 1835 –1842. ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2006.02.004. URL: http://www.sciencedirect.com/science/article/pii/S0167865506000432.

Piciarelli, C., C. Micheloni, and G. L. Foresti (2008). "Trajectory-Based Anomalous Event Detection". In: *IEEE Transactions on Circuits and Systems for Video Technology* 18.11, pp. 1544–1554. DOI: 10.1109/TCSVT.2008.2005599.

Pittore, M., C. Basso, and A. Verri (1999). "Representing and recognizing visual dynamic events with support vector machines". In: *Proceedings 10th International Conference on Image Analysis and Processing*, pp. 18–23. DOI: 10.1109/ICIAP.1999.797565.

Popoola, O. P. and K. Wang (2012). "Video-Based Abnormal Human Behavior Recognition—A Review". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.6, pp. 865–878. DOI: 10.1109/TSMCC.2011.2178594.

Ravanbakhsh, Mahdyar et al. (2016). "Plug-and-Play CNN for Crowd Motion Analysis: An Application in Abnormal Event Detection". In: *CoRR* abs/1610.00307. arXiv: 1610.00307. URL: http://arxiv.org/abs/1610.00307.

Ravanbakhsh, Mahdyar et al. (2017). "Abnormal Event Detection in Videos using Generative Adversarial Nets". In: *CoRR* abs/1708.09644. arXiv: 1708.09644. URL: http://arxiv.org/abs/1708.09644.

Reddy, V., C. Sanderson, and B. C. Lovell (2011). "Improved anomaly detection in crowded scenes via cell-based analysis of foreground speed, size and texture". In: *CVPR 2011 WORKSHOPS*, pp. 55–61. DOI: 10.1109/CVPRW.2011.5981799.

Ribeiro, Manassés, André Eugênio Lazzaretti, and Heitor Silvério Lopes (2018). "A study of deep convolutional auto-encoders for anomaly detection in videos". In: *Pattern Recognition Letters* 105, pp. 13–22.

Roshtkhari, M. J. and M. D. Levine (2013). "Online Dominant and Anomalous Behavior Detection in Videos". In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2611–2618. DOI: 10.1109/CVPR.2013.337.

Ruff, Lukas et al. (2018). "Deep one-class classification". In: *International conference on machine learning*, pp. 4393–4402.

Rumelhart, David E, Geoffrey E Hinton, and Ronald J Williams (1986). "Learning representations by back-propagating errors". In: *nature* 323.6088, pp. 533–536.

Sabokrou, M. et al. (2017). "Deep-Cascade: Cascading 3D Deep Neural Networks for Fast Anomaly Detection and Localization in Crowded Scenes". In: *IEEE Transactions on Image Processing* 26.4, pp. 1992–2004. DOI: 10.1109/TIP.2017.2670780.

Sabokrou, Mohammad, Mahmood Fathy, and Mojtaba Hoseini (2016). "Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder". In: *Electronics Letters* 52.13, pp. 1122–1124.

Sabokrou, Mohammad et al. (2015). "Real-time anomaly detection and localization in crowded scenes". In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 56–62.

Sabokrou, Mohammad et al. (2016). "Fully Convolutional Neural Network for Fast Anomaly Detection in Crowded Scenes". In: *CoRR* abs/1609.00866. arXiv: 1609.00866. URL: http://arxiv.org/abs/1609.00866.

Sabokrou, Mohammad et al. (2018a). "AVID: Adversarial Visual Irregularity Detection". In: *CoRR* abs/1805.09521. arXiv: 1805.09521. URL: http://arxiv.org/abs/1805.09521.

— (2018b). "AVID: Adversarial Visual Irregularity Detection". In: *CoRR* abs/1805.09521. arXiv: 1805.09521. URL: http://arxiv.org/abs/1805.09521.

Sabokrou, Mohammad et al. (2018c). "Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes". In: *Computer Vision and Image Understanding* 172, pp. 88–97.

Saligrama, V and Z Chen (2012a). "Chaotic invariants based on local statistical aggregates". In: *journal=" IEEE Conference on Computer Vision Pattern Recognition*, pp. 2112–2119.

Saligrama, Venkatesh and Zhu Chen (2012b). "Video anomaly detection based on local statistical aggregates". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2112–2119.

Schölkopf, Bernhard et al. (July 2001). "Estimating the Support of a High-Dimensional Distribution". In: *Neural Comput.* 13.7, 1443–1471. ISSN: 0899-7667. DOI: 10.1162/089976601750264965. URL: https://doi.org/10.1162/089976601750264965.

Sermanet, Pierre et al. (2013). "Overfeat: Integrated recognition, localization and detection using convolutional networks". In: *arXiv preprint arXiv:1312.6229*.

Sharif, Md. Haidar, Sahin Uyaver, and Chabane Djeraba (2010). "Crowd Behavior Surveillance Using Bhattacharyya Distance Metric". In: *Computational Modeling of Objects Represented in Images*. Ed. by Reneta P. Barneva et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 311–323. ISBN: 978-3-642-12712-0.

Sharif Razavian, Ali et al. (2014). "CNN Features Off-the-Shelf: An Astounding Baseline for Recognition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Shi, Xingjian et al. (2015). "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting". In: *CoRR* abs/1506.04214. arXiv: 1506.04214. URL: http://arxiv.org/abs/1506.04214.

Simonyan, Karen and Andrew Zisserman (2014). "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556*.

Sodemann, A. A., M. P. Ross, and B. J. Borghetti (2012). "A Review of Anomaly Detection in Automated Surveillance". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.6, pp. 1257–1272. DOI: 10.1109/TSMCC.2012.2215319.

Sun, Jiayu, Jie Shao, and Chengkun He (2019). "Abnormal event detection for video surveillance using deep one-class learning". In: *Multimedia Tools and Applications* 78.3, pp. 3633–3647.

Szegedy, C. et al. (2015). "Going deeper with convolutions". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9. DOI: 10.1109/CVPR.2015.7298594.

Tai Sing Lee (1996). "Image representation using 2D Gabor wavelets". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18.10, pp. 959–971. DOI: 10.1109/34.541406.

Taigman, Y. et al. (2014). "DeepFace: Closing the Gap to Human-Level Performance in Face Verification". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708. DOI: 10.1109/CVPR.2014.220.

Tang, Y. p., X. j. Wang, and H. f. Lu (2009). "Intelligent Video Analysis Technology for Elevator Cage Abnormality Detection in Computer Vision". In: *2009 Fourth International Conference on Computer Sciences and Convergence Information Technology*, pp. 1252–1258. DOI: 10.1109/ICCIT.2009.206.

Tao Xiang and Shaogang Gong (2005). "Video behaviour profiling and abnormality detection without manual labelling". In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Vol. 2, 1238–1245 Vol. 2. DOI: 10.1109/ICCV.2005.248.

Toshev, Alexander and Christian Szegedy (2013). "DeepPose: Human Pose Estimation via Deep Neural Networks". In: *CoRR* abs/1312.4659. arXiv: 1312.4659. URL: http://arxiv.org/abs/1312.4659.

Utasi, Ákos and László Czúni (2010). "Detection of unusual optical flow patterns by multi-level hidden Markov models". In: *Optical Engineering* 49.1, p. 017201.

Utasi, Ákos and László Czúni (2010). "Detection of unusual optical flow patterns by multi-level hidden Markov models". In: *Optical Engineering* 49.1, pp. 1 –11. DOI: 10.1117/1.3280284. URL: https://doi.org/10.1117/1.3280284.

Vapnik, V. (1963). "Pattern recognition using generalized portrait method". In: *Automation and Remote Control* 24, pp. 774–780.

Vapnik, Vladimir N. (2000). "Introduction: Four Periods in the Research of the Learning Problem". In: *The Nature of Statistical Learning Theory*. New York, NY: Springer New York, pp. 1–15. ISBN: 978-1-4757-3264-1. DOI: 10.1007/978-1-4757-3264-1_1. URL: https://doi.org/10.1007/978-1-4757-3264-1_1.

Wang, L. and D. Suter (2007). "Recognizing Human Activities from Silhouettes: Motion Subspace and Factorial Discriminative Graphical Model". In: *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. DOI: 10.1109/CVPR.2007.383298.

Wang, T. and H. Snoussi (2012). "Histograms of Optical Flow Orientation for Visual Abnormal Events Detection". In: *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, pp. 13–18. DOI: 10.1109/AVSS.2012.39.

— (2014). "Detection of Abnormal Visual Events via Global Optical Flow Orientation Histogram". In: *IEEE Transactions on Information Forensics and Security* 9.6, pp. 988–998. DOI: 10.1109/TIFS.2014.2315971.

Wang, Tian et al. (2018). "Generative neural networks for anomaly detection in crowded scenes". In: *IEEE Transactions on Information Forensics and Security* 14.5, pp. 1390–1399.

Wu, Shandong, Brian E Moore, and Mubarak Shah (2010). "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes". In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2054–2060.

Wu, Yonghui et al. (2016). "Google's neural machine translation system: Bridging the gap between human and machine translation". In: *arXiv preprint arXiv:1609.08144*.

Xiang, T., S. Gong, and D. Parkinson (2002). "Autonomous Visual Events Detection and Classification without Explicit Object-Centred Segmentation and Tracking". In: *Proceedings of the British Machine Vision Conference*. doi:10.5244/C.16.2. BMVA Press, pp. 21.1–21.10. ISBN: 1-901725-19-7.

Xiao, T., C. Zhang, and H. Zha (2015). "Learning to Detect Anomalies in Surveillance Video". In: *IEEE Signal Processing Letters* 22.9, pp. 1477–1481. DOI: 10.1109/LSP.2015.2410031.

Xiao, Tan, Chao Zhang, and Hongbin Zha (2015). "Learning to detect anomalies in surveillance video". In: *IEEE Signal Processing Letters* 22.9, pp. 1477–1481.

Xu, Dan et al. (2017). "Detecting anomalous events in videos by learning deep representations of appearance and motion". In: *Computer Vision and Image Understanding* 156. Image and Video Understanding in Big Data, pp. 117 –127. ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2016.10.010. URL: http://www.sciencedirect.com/science/article/pii/S1077314216301618.

Xu, Jie, Kanmin Xue, and Kang Zhang (2019). "Current status and future trends of clinical diagnoses via image-based deep learning". In: *Theranostics* 9.25, p. 7556.

Zelnik-Manor, L. and M. Irani (2006). "Statistical analysis of dynamic actions". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.9, pp. 1530–1535. DOI: 10.1109/TPAMI.2006.194.

Zhang, Chengcui et al. (2010). "A Multiple Instance Learning and Relevance Feedback Framework for Retrieving Abnormal Incidents in Surveillance Videos". In: *J. Multim.* 5.4, pp. 310–321. DOI: 10.4304/jmm.5.4.310-321. URL: https://doi.org/10.4304/jmm.5.4.310-321.

Zhang, Xiangyu et al. (2015). "Accelerating very deep convolutional networks for classification and detection". In: *IEEE transactions on pattern analysis and machine intelligence* 38.10, pp. 1943–1955.

Zhao, B., L. Fei-Fei, and E. P. Xing (2011). "Online detection of unusual events in videos via dynamic sparse coding". In: *CVPR 2011*, pp. 3313–3320. DOI: 10.1109/CVPR.2011.5995524.

Zhao, Yiru et al. (2017). "Spatio-Temporal AutoEncoder for Video Anomaly Detection". In: *Proceedings of the 25th ACM International Conference on Multimedia*. MM '17. Mountain View, California, USA: Association for Computing Machinery, 1933–1941.

ISBN: 9781450349062. DOI: 10.1145/3123266.3123451. URL: https://doi.org/10.1145/3123266.3123451.

Zhou, J. T. et al. (2019). "AnomalyNet: An Anomaly Detection Network for Video Surveillance". In: *IEEE Transactions on Information Forensics and Security* 14.10, pp. 2537–2550. DOI: 10.1109/TIFS.2019.2900907.

Zhou, Shifu et al. (2016a). "Spatial–temporal convolutional neural networks for anomaly detection and localization in crowded scenes". In: *Signal Processing: Image Communication* 47, pp. 358–368.

— (2016b). "Spatial–temporal convolutional neural networks for anomaly detection and localization in crowded scenes". In: *Signal Processing: Image Communication* 47, pp. 358–368. ISSN: 0923-5965. DOI: https://doi.org/10.1016/j.image.2016.06.007. URL: http://www.sciencedirect.com/science/article/pii/S0923596516300935.

Zhou, Shuchang et al. (2017). "GeneGAN: Learning Object Transfiguration and Attribute Subspace from Unpaired Data". In: *CoRR* abs/1705.04932. arXiv: 1705.04932. URL: http://arxiv.org/abs/1705.04932.

Slim HAMDI

Doctorat : Optimisation et Sûreté des Systèmes

Année 2021

Détection d'anomalies par apprentissage profond pour la surveillance par drone

La sécurité civile est l'ensemble des moyens mis en œuvre par un État ou une organisation pour protéger les populations civiles, ainsi que leurs biens et activités, en temps de guerre, de crise et de paix, contre les risques ou menaces de toute nature. En outre, elle consiste à assurer la sécurité des personnes contre tous types de risques naturels tels que les incendies ou contre diverses menaces pouvant mettre en danger leur vie, ainsi que celle de leurs biens ou activités (actes de terrorisme, actes de vandalisme, etc.). Ces dernières années, l'utilisation de drones pour des tâches de surveillance s'est développée dans le monde entier. Ainsi, le nombre de caméras qui doivent être analysées augmente et l'efficacité et la précision des opérateurs humains ont atteint leurs limites. De plus, dans le contexte de la détection d'anomalies, seuls les événements normaux sont disponibles pour le processus d'apprentissage. Par conséquent, la mise en œuvre d'une méthode d'apprentissage profond en mode non supervisé pour résoudre ce problème devient fondamentale. Dans cette thèse, nous avons proposé plusieurs architectures d'apprentissage profond capables de détecter des événements anormaux avec des performances élevées.

Mots clés : détection des anomalies (informatique) – apprentissage profond – transfert d'apprentissage.

Deep Learning Anomaly Detection for Drone-based Surveillance

Civil security is the set of methods implemented by a State or an organization to protect civilian populations, as well as their property and activities, in times of war, crisis, and peace, against risks or threats of any kind. Moreover, it consists of ensuring the safety of people against all types of natural risks such as fires or against various threats that could endanger their lives, as well as that of their property or activities (acts of terrorism, acts of vandalism, etc.). In recent years, the use of drones for surveillance tasks has been on the rise worldwide. So, The number of cameras that must be analyzed increases and the efficiency and accuracy of human operators have reached their limits. Moreover, in the context of anomaly detection, only normal events are available for the learning process. Therefore, the implementation of a deep learning method in unsupervised mode to solve this problem becomes fundamental. In this thesis, we have proposed many deep learning architectures capable of detecting abnormal events with high performance.

Keywords: anomaly detection (computer security) – deep learning – transfer of training.

Thèse réalisée en partenariat entre :

Ecole Doctorale "Sciences pour l'Ingénieur"