Simultaneously Spatio-temporal Driving Motion Anomaly Detection by Multi-Layer Isolation-based Sifter

LOTVS

Abstract

Driving anomaly detection in dashcam videos begin to draw research interests recently in advanced driver assistance systems, including self-driving cars. Commonly, motion clues are usually utilized in these works. The motion of anomaly in dashcam videos has more ambiguities than that of surveillance videos, which hinders to train a supervised discriminator. To address this problem, we formulate an *unsupervised multi-layer sifting process* fulfilled by a simple but effective Multi-layer Isolation-based Sifter (MLayer-iSifter), which consists of alternatively spatial-temporal *pruning* layers and *isolation forest decision* layers. The behind idea is to eliminate the driving motion ambiguity in the dynamic traffic scenes by gradually improving the discriminative ability of the sifted normal motion patterns, and this is successfully fulfilled by increasing the number of layers of MLayer-iSifter. This paper represents the motion of videos by aggregating multiple conv-layers of a pre-trained CNN model coupling a temporal consistency measurement. The obtained spatio-temporal motion representations of videos are then fed into MLayer-iSifter to simultaneously find the spatial anomaly regions and temporal anomaly frames. Extensive experimental results on a dataset with 106 videos manually annotated carefully by ourselves demonstrate the favorable performance.

Introduction

More and more research efforts have been put on advanced driving systems in recent years, including the self-driving cars, and the research purpose becomes pursuing safer, agiler and more dexterous driving experiences than ever. In order to make intelligent vehicles respond to various driving situations/scenarios smart and immediately, unexpected driving anomalies should be paid more attention for truly safe driving compared with the normal driving situations.

Generally, detecting driving behaviour anomaly in dashcam videos becomes difficult especially because of the drastic camera motion and chaotic background, and these issues further cause large ambiguities between anomaly and normal driving behaviours. Recent research in computer vision has began to address this problem from different views. For instance, Kataoka *et al.* (Kataoka et al. 2018a; 2018b) and Chan *et al.* (Chan et al. 2016) anticipated the traffic accident through adaptive loss and dynamic-spatial-attention (DSA) recurrent neural network (RNN), respectively. Yuan *et al.* (Yuan, Fang, and Wang 2018) addressed the driving anomaly by incremental motion consistency measurement. However, these methods demonstrate at least two disadvantages. 1) They focus on predicting the future accident based on the historical observations, and concentrate on the moving pedestrians or vehicles pre-detected. However, many kinds of driving anomalies occur suddenly and unpredictable, and pre-trained detectors can not cover all abnormal situations in traffic scenes, such as the tire blowout and bombing of cars, shown in Fig. 1(a) and Fig. 1(b), respectively. 2) The anomaly motion in dashcam videos has large ambiguity with dynamic background and other moving objects, which hinders to train a supervised classifier.

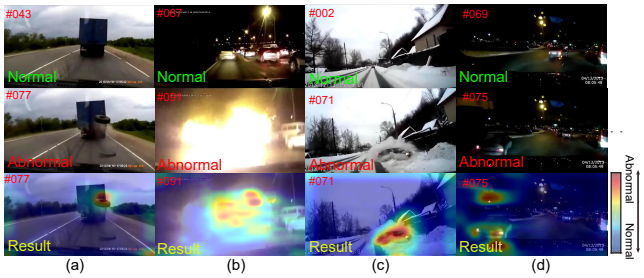

Figure 1: Some typical situations from normal (the first row) to abnormal (the second row) and the detected spatial anomaly confidence maps by the proposed method (the third row). The anomaly situations are (a) tire blowout, (b) sudden bombing of a car, (c) a car is rushing into the snowy roadside, and (d) a truck suddenly hits a car. Note that, it is difficult to detect pedestrians or vehicles in these situations, and these driving anomalies are unpredictable without any omens.

In this paper, we propose an unsupervised approach to detect the spatial-temporal anomaly within many unlabeled dashcam videos by learning the normal motion representation from themselves, and we believe a good normal motion representation in driving is a big help for driving motion anomaly detection. Therefore, we achieve normal motion representation of driving by making a trade-off between two conflictive goals: boosting its compactness while improving its discriminative ability. These two goals usually are conflictive because more compact normal patterns may have weaker discriminative ability since it cannot represent large variations of the dataset. We address this issue through an

unsupervised multi-layer sifting process. This process is fulfilled through a simple yet effective Multi-layer Isolation-based Sifter (MLayer-iSifter) which consists of alternatively spatial-temporal pruning layers and isolation forest decision layers. We formulate the driving motion anomaly detection as a two-stage procedure: *normal motion sifting of group videos* (i.e., normal driving motion representation) and *abnormal motion sifting of individual video* (i.e., abnormal driving motion discrimination).

In particular, we adopt a promising multi-layer-convolutional feature map aggregation (Yu, Wang, and Darrell 2018) to represent the spatial motion of the video and aggregate them in frame-level. Inspired by that the feature maps of a deep-convolutional layer represent the *part-proposals* (Xu et al. 2018) of the image, the spatial anomaly regions can be directly obtained from inferring the anomaly of these part-proposals. The temporal consistency of the spatial-temporal motion is measured with a Gaussian process regressor (GPR) (Ounpraseuth 2006), where each frame's motion consistency within a temporal interval is considered for restraining the motion estimation error and manifesting the large temporal variation which potentially contains anomaly.

The **contributions** of this work are twofold.

(1) The MLayer-iSifter architecture can eliminate the motion ambiguity of driving anomaly in dashcam videos by increasing the number of of layers, and *simultaneously* find out the reliable spatial-temporal driving anomaly *efficiently*.

(2) A new dataset containing 106 video clips (100 frames/clip) was constructed. We manually labeled both temporal anomaly frames and spatial anomaly regions of all the video clips.

Extensive experimental results demonstrate that our method can detect spatial-temporal driving motion anomaly much more accurately than the state-of-the-arts.

Related Work

Video anomaly is commonly defined as the target behaviour with rarity occurrence, dissimilar pattern with pre-defined normal model/rules, and deviated context (Chandola, Banerjee, and Kumar 2009; Giorno, Bagnell, and Hebert 2016). Many efforts are devoted to normal behaviour modeling, spatial and temporal consistency/dependency measurement of behaviours explored by anomaly detection in surveillance systems and advanced driving systems, including self-driving cars (Zhang et al. 2017).

Normal Behaviour Modeling. For modeling the normal behavior, exploiting the normal rules contained in the trajectories is a standard approach (Laxhammar and Falkman 2014; Jiang, Wu, and Katsaggelos 2009), which can capture the long-term semantics of objects while often fails to track accurately because of various disturbing factors, e.g., occlusion, fast motion, similar object surrounded, and so on. Alternatively, recent approaches unitized the hand-craft low-level features (e.g., HOG, HOF, STIPs, etc.) extracted from 2D region(s) or 3D volumes. Commonly, these locally low-level features are feeded into various detectors trained by normal samples, such as distance-based (Cheng, Chen, and Fang 2015), sparse-coding (Zhao, Li, and Xing 2011; Luo, Liu, and Gao 2017), domain-based (one class SVM) (Chen, Qian, and Saligrama 2013), probabilistic-based (e.g., mixture of probabilistic PCA (MPPCA) (Kim and Grauman 2009) and Gaussian process regressor (Cheng, Chen, and Fang 2015)), Graph-based inference machines (Liu, Ting, and Zhou 2012) and physical-inspired models (Mehran, Oyama, and Shah 2009). Some recent models adopted autoencoders (Tran and Hogg 2017) to learn deep features, expressive CNNs (Sabokrou et al. 2016; 2017) or predictive RNNs (Chan et al. 2016; Kiran, Thomas, and Parakkal 2018) by minimizing the reconstruction/expression/prediction error of the input samples. In relative to this paper, Chan *et al.* (Chan et al. 2016) proposed a dynamic-spatial-attention (DSA) recurrent neural network (RNN) for anticipating accidents in dashcam videos. Most of normal behaviour modeling methods need plenty of annotated training data, while it is difficult to discriminate the abnormal against normal motion in driving scenarios.

Spatial and Temporal Consistency Measurement. Spatial-temporal consistency is mainly inspired by the co-occurrence of appearance or motion patterns in spatial regions over time, and filters the local anomaly scores obtained by normal discriminators (e.g., hidden Markov model (HMM) (Kratz and Nishino 2009), Gaussian mixture model (GMM) (Basharat, Gritai, and Shah 2008), mixture of dynamic texture (MDT) (Li, Mahadevan, and Vasconcelos 2014), Gaussian process regression (GPR) (Cheng, Chen, and Fang 2015)). For example, Yuan *et al.* (Yuan, Fang, and Wang 2015) utilized spatial-temporal context consistency of pedestrians to conduct the crowd anomaly, and had addressed the driving anomaly by motion consistency (Yuan, Fang, and Wang 2018). For representing the spatial-temporal consistency, some works embedded the high-level structured consistency for anomaly detection in videos, such as the feature grouping of individuals by manifold learning (Rao et al. 2016). Additionally, because of the great success of CNN or RNN approaches in many visual tasks, the latest approaches checked the spatial-temporal consistency by exploiting the dependency of the behaviors between frames, such as LSTM predictor (Kiran, Thomas, and Parakkal 2018) and sequential generator (Liu et al. 2018). For instance, Liu *et al.* (Liu et al. 2018) leveraged a future frame prediction based framework for anomaly detection by generative adversarial network (GAN). These aforementioned methods usually treat the spatial and temporal anomaly in two separate stages, which may easily cause mistaken detections if the output of the first-stage contain errors.

Proposed Method

The multi-layer sifting process includes two stages: normal motion sifting of group videos and abnormal motion sifting of individual video. In each sifting process, there is one common preliminary component: spatial-temporal motion aggregation. The architecture of the proposed method is illustrated in Fig. 2. We will elaborate its two main components: 1) spatial-temporal motion aggregation, and 2) driving motion anomaly detection which is implemented by a multi-layer isolation-based sifter (MLayer-iSifter).

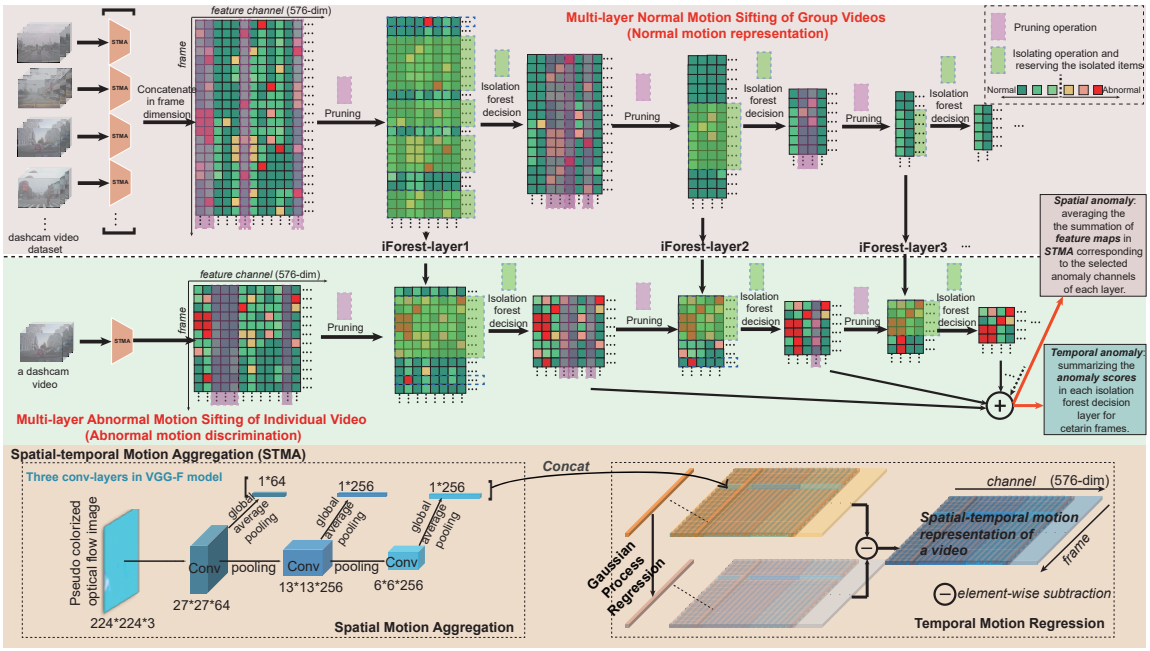

Figure 2: The architecture of the proposed method. The first row is the procedure of normal motion sifting of group videos. The second row is the abnormal motion sifting process of individual video, and the spatial-temporal motion aggregation is presented in the bottom row. The spatial driving motion anomaly is determined by averaging the summation of the selected feature maps (corresponding to abnormal feature channels) in each layer, which is designed to take out the true anomaly and depressing the false-alarm in different number of layers. Temporal driving motion anomaly is obtained by summarizing the anomaly scores after each forest decision layer for certain frames.

Spatial-temporal Motion Aggregation (STMA)

We first estimate the motion of each two adjacent frames (In this paper, an efficient method (Liu 2009) is used). Here, instead of using histogram to represent the optical flow, we feed the pseudo colorized optical flow image through a pre-trained CNN model to obtain the spatial motion representation. We hierarchically aggregate multiple convolutional layers (it is denoted as conv-layers distinguishing from the layer term used in MLayer-iSifter.) of feature maps. Because traffic scenes are dynamic, we further introduce the Gaussian process regressor to restrain the motion estimation error of optical flow and measure the temporal motion consistency within a temporal interval. The spatial-temporal motion aggregation is divided into spatial motion aggregation and temporal motion regression.

Spatial Motion Aggregation. Given a video V with F frames, we first compute the optical flow M_i of the i^{th} frame of V. Then we pass M_i through a pre-trained CNN model and obtain multiple convolutional feature maps $\{\mathbf{F}_{i,1} \in \mathbb{R}^{W_1 \times H_1 \times C_1}, ..., \mathbf{F}_{i,k} \in \mathbb{R}^{W_k \times H_k \times C_k}, ..., \mathbf{F}_{i,K} \in \mathbb{R}^{W_K \times H_K \times C_K}\}$, where $k \in [1, K]$ is the conv-layer number, and W_k, H_k and C_k are the width, height and channels of feature maps in the k^{th} convolutional layer. These tensors are transformed into vectors by global average pooling (Zhou et al. 2016):

$$\mathbf{v}_{i,k} = \frac{1}{H_k * W_k} \sum_{H_i} \sum_{W_i} \mathbf{F}_{i,k}. \tag{1}$$

The motion representation of the i^{th} frame is concatenated as $\mathbf{v}_i$=$[\mathbf{v}_{i,1}, \mathbf{v}_{i,2}, ..., \mathbf{v}_{i,k}] \in \mathbb{R}^{1 \times \sum C_k}$, and the spatial-temporal motion representation of video V with F frames is denoted as $\mathbf{V} = [\mathbf{v}_1, ..., \mathbf{v}_F] \in \mathbb{R}^{F \times \sum C_k}$. The weight for convolutional filters over frames are shared.

Temporal Motion Regression. Gaussian process regressor aims to compute the residual between the spatial motion representation of a frame with its regressed value. This consideration involves two underlying factors. First, the convolutional filters in different frames are shared where each filter focuses on the same semantic parts within the frames, i.e., the feature channels over frames have temporal consistency.Second, there may be estimation error in optical flow which should be restrained, and the temporal variation potentially caused by anomaly needs to be manifested.

The formulation of Gaussian process regressor follows Gaussian process regression (GPR), but without any supervising label. To be specific, for the spatial-motion representation $\mathbf{V} \in \mathbb{R}^{F \times \sum C_k}$, this work treats the frame indexes as the observed data $\mathbf{x} \in \mathbb{R}^{F \times 1} = [x_1, ..., x_f, ..., x_F]$, where f is the frame index, and the c^{th} column of $\mathbf{V}$ as the predicted label $\mathbf{y}_c \in \mathbb{R}^{F \times 1}$, where c is the channel index of $\mathbf{V}$ and $c \in [1, \sum C_k]$. The predicted error $\mathbf{d}_c$ is denoted as:

$$\mathbf{d}_c = \mathbf{y}_c - \mathbf{y}_c^*, \tag{2}$$

where $\mathbf{y}_c^*$ is computed by GPR:

$$p(\mathbf{y}_c^* | \mathbf{x}, \mathbf{y}_c, \mathbf{x}^*) \sim \mathcal{N}(\bar{\mathbf{y}}_c^*, \Sigma \mathbf{y}_c^*), \tag{3}$$

where $\mathbf{x}^*$ and $\mathbf{y}_c^*$ denote the testing data and its corresponding label, respectively. Note that $\mathbf{x}$=$\mathbf{x}^*$ in our work, i.e., the frame indexes. $\mathbf{y}_c^* = \mathbf{K}_*^T(\mathbf{K} + \sigma_n^2 \mathbf{I}_n)^{-1} \mathbf{y}_c$ and $\mathbf{y}_c^* = \mathbf{K}_{**} - \mathbf{K}_*^T(\mathbf{K} + \sigma_n^2 \mathbf{I}_n)^{-1} \mathbf{K}_*$, where $\mathbf{I}_n$ is an identity matrix, and $\mathbf{K}_{**}$, $\mathbf{K}_*$ and $\mathbf{K}$ are the covariance matrixes denoted by $k(\mathbf{x}^*, \mathbf{x}^*)$, $k(\mathbf{x}, \mathbf{x}^*)$ and $k(\mathbf{x}, \mathbf{x})$. They are the

same and can be computed once in this paper. $k(x_f, x'_f)$ is the covariance function evaluated based on the radial basis function (RBF) kernel (Ounpraseuth 2006).

$$k(x, x') = \sigma_V^2 \exp(-\frac{(x_f - x'_f)^2}{2l^2}) + \sigma_n^2 \delta(x_f - x'_f), \quad (4)$$

where $x_f \in \mathbf{x}$, $x_f \in \mathbf{x}^*$, and $\delta(x_f - x'_f)$ is the Kronecker delta. It is worthy noting that hyper-parameters in RBF kernel include the scale l, the signal variance σ_V^2, and the noise variance σ_n^2, and they can be determined by maximizing the marginal likelihood of the observed data with the conjugate gradient method.

Consequently, we conduct Gaussian process regression on each column of $\mathbf{V}$, and transform it as $\mathbf{D} \in \mathbb{R}^{F \times \sum C_k} = [\mathbf{d}_1, ..., \mathbf{d}_c, ..., \mathbf{d}_{\sum C_k}]$. Actually, $\mathbf{D}$ represents the residual between the aggregated spatial motion representations of the frames and their regressed values by consistency measurement in a temporal interval, and we found this consideration can largely boost the performance.

Driving Motion Anomaly Detection

Assume there are N available dashcam videos. Their motion representations after STMA is defined as $\mathcal{D} = \{\mathbf{D}_1, ..., \mathbf{D}_n, ..., \mathbf{D}_N\}$. As aforementioned, in order to detect the driving motion anomaly within $\mathbf{D}_n$, we need to learn the normal motion representations of driving in $\mathcal{D}$.

Normal Motion Sifting of Group Videos. This procedure is through a multi-layer architecture, and aims to solve the conflictive goals: making the normal representation compact while improving its discriminative ability by increasing the number of layers. It contains alternatively spatial-temporal pruning layers and temporal isolation forest decision layers. We concatenate the spatial-temporal motion of group videos in frame-level, i.e., rearranging $\mathcal{D}$ as a matrix $\mathbf{W} = [\mathbf{D}_1^T, ..., \mathbf{D}_n^T, ..., \mathbf{D}_N^T]^T \in \mathbb{R}^{\sum_{n=1}^{N} F_n \times \sum C_k}$, which denotes the input of the first layer of the multi-layer architecture, where F_n is the frame number of $\mathbf{D}_n$. Assume $\mathbf{W}$ is changed as $\mathbf{W}_l \in \mathbb{R}^{P_l \times Q_l}$ in the l^{th} layer, where P_l and Q_l are the corresponding number of frames and channels, respectively ($P_1 = \sum_{n=1}^{N} F_n$, $Q_1 = \sum C_k$).

Spatial-temporal pruning layer: This layer aims to select the normal *part proposals* (Xu et al. 2018), i.e., the feature channels responding to the normal motion parts in frames. In other words, this layer obtains a subset $\mathbf{E}_l \subset \mathbf{W}_l$ after pruning some channels (denoting as $\mathbf{E}'_l$) of $\mathbf{W}_l$. The pruning layer conducts an operation of:

$$\mathbf{E}_l = \mathbf{W}_l \backslash \mathbf{E}'_l. \quad (5)$$

For this operation, this paper evaluates the variance of each column of $\mathbf{W}_l$ because frequent appearing and disappearing of objects within a video will cause large variances over temporal frames. On the contrary, the stable and normal motion parts of the scene over frames have smaller variance, and vice versa for anomaly motion parts. To be specific, the set of variance $V(\mathbf{W}_l) = [v_{1_l}, ..., v_{q_l}, ..., v_{Q_l}]$ of $\mathbf{W}_l$ over frames is computed by:

$$v_{q_l} = \frac{1}{P_l} \sum_{f=1}^{P_l} (g_{p_l}^{q_l} - \bar{g}^{q_l})^2, \quad (6)$$

where $g_{p_l}^{q_l}$ is the motion value in the ${p_l}^{th}$ row of the ${q_l}^{th}$ column of $\mathbf{W}_l$, in which $p_l \in [1, P_l]$. $\bar{g}^{q_l}$ is the average value of $g_{p_l}^{q_l}$. Then we sort $V(\mathbf{W}_l)$ as $\tilde{V}(\mathbf{W}_l)$ in **ascending** order and pick $\mathbf{E}_l$ by:

$$\mathbf{E}_l = \{\mathbf{w}_{q_l} \in \mathbf{W}_l | v_{q_l} < \eta_l\}, \quad (7)$$

where $\mathbf{w}_{q_l}$ is the ${q_l}^{th}$ column vector of $\mathbf{W}_l$, η_l is a threshold for separating the motion into normal motion parts and abnormal ones. Here, η_l is set as the $(\frac{\sum C_k}{\alpha})^{th}$ value in $\tilde{V}(\mathbf{W}_l)$, where α determines the capacity of normal sifting in each layer. The larger the α is, the more compact the sifted normal motion patterns are, whereas the weaker discriminative ability the normal motion patterns have, and vice versa. In this paper, we empirically set α as 2. The pruning layer makes the spatial motion part compact with the help of their temporal variances.

Temporal isolation forest decision layer: The pruning layer aims to make the normal motion patterns compact, while this layer instead is to improve the discriminative ability of the sifted normal motion patterns. In this layer, we find a subset $\mathbf{W}_{l+1} \subset \mathbf{E}_l$ in $\mathbf{E}_l$ that is closer to anomaly than the remaining items.

For this purpose, this paper introduces the isolation forest (iForest) (Liu, Ting, and Zhou 2012) because of its linear complexity, effectiveness and minimum memory-requirement. Isolation forest, containing λ isolation trees (iTrees), is constructed by recursively partitioning the sub-instance set with size s (a part of the whole instance set) on each tree until all of them become a singleton, and resulting in proper binary tree set such that the number of nodes of a tree is $2s - 1$, where the instance is commonly a vector. The outlier score of an instance $\mathbf{x}$ is determined by averaging its path length $p(\mathbf{x})$ on each iTree:

$$A(\mathbf{x}) = 2^{-\overline{p(\mathbf{x})}/c(s)}, \quad (8)$$

where $c(s) = 2H(s-1) - 2(s-1)/\gamma$, γ is the total instance number, H specifies the harmonic number, and $\overline{P(\mathbf{x})}$ is the average path length of $\mathbf{x}$ on λ iTrees.

This layer conducts the following operations:

$$\begin{aligned} iForest_l &= \texttt{iForest}_{train}(\mathbf{E}_l, \lambda, s), \\ A(\mathbf{E}_l) &= \texttt{iForest}_{test}(iForest_l, \mathbf{E}_l), \\ \mathbf{W}_{l+1} &= \{\mathbf{e}_{p_l} \in \mathbf{E}_l | A(\mathbf{e}_{p_l}) > \overline{A(\mathbf{E}_l)}\}, \end{aligned} \quad (9)$$

where $\mathbf{e}_{p_l}$ is the p_l^{th} row vector of $\mathbf{E}_l$, $\texttt{iForest}_{train}(\cdot)$ and $\texttt{iForest}_{test}(\cdot)$ are the procedures of training the iForest with instances and evaluating the anomaly of instances, respectively. $iForest_l$ denotes the encoded iForest in l^{th} layer, which will be utilized to determine anomaly, and $\overline{A(\mathbf{E}_l)}$ is the average anomaly score over all rows in $\mathbf{E}_l$. For λ and s, we follow the setting strategy of (Liu, Ting, and Zhou 2012), i.e., $\lambda = 100$ and $s = 256$ in our experiments.

In order to achieve a compact representation of normal patterns, we feed $\mathbf{W}_{l+1}$ to the next pruning layer, and repeat this until *only one feature channel is left*. Assume L layers are generated after this stage.

Abnormal Motion Sifting of Individual Video. This procedure is also a multi-layer architecture. The structure is

similar to normal motion sifting but with modifications for pruning layer and isolation forest decision layer. The temporal anomaly frames and spatial anomaly regions are *simultaneously* localized in this stage.

For the spatial-temporal motion representation of the n^{th} video $\mathbf{D}_n \in \mathbb{R}^{F_n \times \sum C_k}$, assume it is changed into $\mathbf{O}_l$ by passing it into the following alternative pruning layers and isolation forest decision layers.

Spatial-temporal pruning layer: Similar to normal motion sifting, this layer also computes the variance of each column of $\mathbf{O}_l$ by Eq. 6. We denote the variance set as $V(\mathbf{O}_l)$. Differently, we sort $V(\mathbf{O}_l)$ as $\tilde{V}(\mathbf{O}_l)$ in **descending** order, and pick the subset $\mathbf{Q}_l \subset \mathbf{O}_l$ whose column variances are larger than the $(\frac{\sum C_k}{2})^{th}$ value of $\tilde{V}(\mathbf{O}_l)$.

Isolation forest decision layer: We compute the anomaly scores of frames in $\mathbf{Q}_l$ by the encoded $iForest_l$ in normal motion sifting process, i.e.,

$$\begin{aligned} A(\mathbf{Q}_l) &= \texttt{iForest}_{test}(iForest_l, \mathbf{Q}_l), \\ \mathbf{O}_{l+1} &= \{\mathbf{q}_{f_l} \in \mathbf{Q}_l | A(\mathbf{q}_{f_l}) > \overline{A(\mathbf{Q}_l)}\}, \end{aligned} \tag{10}$$

where $\mathbf{q}_{f_l}$ is the ${f_l}^{th}$ row vector of $\mathbf{Q}_l$, and $A(\mathbf{Q}_l)$ is the anomaly score in l^{th} layer over the rows of $\mathbf{Q}_l$. Then $\mathbf{O}_{l+1}$ is fed into the next pruning layer until $l = L$.

Temporal Anomaly Determination. After passing $\mathbf{D}_n$ through L layers for sifting anomaly, the temporal anomaly T_a^f of the $f^{th} \in [1, F_n]$ frame in the n^{th} dashcam video is determined by summarizing the anomaly scores of different number of layers, i.e.,:

$$T_a^f = \sum_{l=1}^{L} A(\mathbf{Q}_l)|_f, \tag{11}$$

Then we re-weight the temporal anomaly score to the range of $[0, 1]$ for $\mathbf{D}_n$ by a min-max normalizer.

Spatial Anomaly Determination. The pruning layer aims to select the anomaly channels of STMA of a video, and the selected channels link the feature maps denoting the part-proposals (Xu et al. 2018). In fact, in each anomaly sifting layer, the selected anomaly part-proposals of a frame reflect its anomaly confidence maps. In order to obtain a reasonable determination, we design a *fusion strategy* which firstly obtains an anomaly confidence map by summarizing the selected anomaly part-proposals in each layer, and then average these anomaly confidence maps of all layers, i.e.,:

$$\mathbf{S}_a^f = \frac{1}{L} \sum_{l=1}^{L} \sum_{b=1}^{B_l} \mathbf{F}_{l,b}^f, \tag{12}$$

where B_l is the left number of channels in the l^{th} layer, and f is the frame index. Note that, the anomaly channels may come from different conv-layers of feature maps. Therefore, we resize $\mathbf{F}_{l,b}^f$ as the original size of image by bilinear interpolation. We normalize $\mathbf{S}_a^f$ by the min-max normalizer. For the whole video, we incorporate the temporal anomaly scores and spatial anomaly maps together by $T_a^f \times \mathbf{S}_a^f$.

Experiments

In this section, firstly, the evaluation metrics and dataset are described. Note that the multi-layer normal motion sifting process is conducted by feeding the spatial-temporal motion representations of all videos in the dataset. Secondly, the performance on different number of layers for abnormal motion sifting is evaluated. Finally, the comparison between the proposed method and five state-of-the-art approaches are presented for validating the superiority, as well as their computational cost analysis.

For the spatial-temporal motion aggregation, we utilize $Conv2$, $Conv5$ and $fc6$ layers of a pre-trained VGG-F model (Chatfield et al. 2014) on ImageNet to represent the multi-scale motion feature per frame. Note that the pre-trained model here also can be replaced by other CNN models. We use the open-source library MatConvNet (Vedaldi and Lenc 2015) for conducting experiments. All the experiments are run on a computer with an Intel i7 CPU and 8G memory.

Dataset and Evaluation Metrics

According to our knowledge, there is no publicly available dataset with a temporal-spatial labeling for driving motion anomaly detection. The most relative one is the crowd-sourced dashcam video dataset for accident anticipation [1] contributed by (Chan et al. 2016), in which each video has 100 frames and the last 10 frames are temporally labeled as anomaly. Apparently, this setting is not universal in practice. In addition, Kataoka *et al.* (Kataoka et al. 2018a; 2018b) constructed an accident video benchmark with 6000 clips. However, the clips are only temporally labeled and not publicized. Hence, we construct a new dataset (DriveAnomaly106) containing 106 video clip (100 frames per clip) which are under various weather condition (day, nightfall, night, snowy, rainy, foggy, etc.) and manual annotated carefully for both temporal and spatial anomaly. Some of them are collected from (Chan et al. 2016). The resolution of the frame is 476×265, and the anomaly regions are masked by their instance-level contours. For anomaly labeling, we adopt two principles: 1) The anomaly is object-oriented and threatening to the ego-vehicle, such as vehicle crossing, overtaking, and so on; 2) The anomaly owns a manifest trend to cause an accident with ego-vehicle or other objects. Spatial anomaly regions and temporal frames are labeled by five volunteers with over ten years of driving experience and their common labeling results are reserved.

For the evaluation metrics, following the video anomaly detection methods (Liu et al. 2018; Giorno, Bagnell, and Hebert 2016), we employ the standard frame-level and pixel-level ROC and area under ROC (AUC) to quantitatively qualify the performance. Specifically, the spatial anomaly detection performance is further evaluated by the ratio of anomaly region detection (RD), as was explained in Li *et al.* (Li, Mahadevan, and Vasconcelos 2014).

Evaluation on Different Number of Layers

Since this work proposes a multi-layer sifting process for abnormal driving motion detection, we will evaluate the performance of of the proposed method with different number of layers. The number of layers in the proposed framework

[1] http://aliensunmin.github.io/project/dashcam/

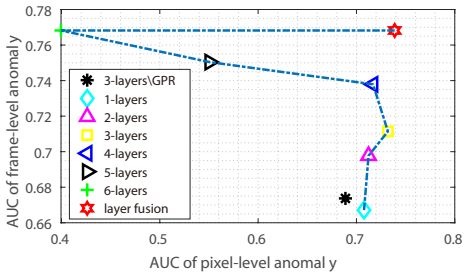

Figure 3: The frame-level anomaly detection AUC values and pixel-level ones w.r.t., different number of layers.

is automatically determined. In the experiment, 6 layers are generated. Fig. 3 shows the pixel-level AUC in conjunction with the frame-level AUC value tendency with different number of layers. Fig. 4 shows some detection results of spatial anomaly regions. These results show that the temporal anomaly detection performance demonstrates a manifest increase by increasing the number of layers, whereas the spatial anomaly detection performance increases till three layers and descends drastically after five and six layers. After checking the reasons, we find that this phenomena is caused by that fewer channels are left in more layers (2 channels left after 5 layers and only one channel left for after 6 layers), where the left anomaly motion parts cannot cover the complete anomaly regions even with an entire zero map. However, the layer fusion strategy can take out the anomaly regions in most configurations of different number of layers, and depress the false-alarm, as shown in Fig. 4. The Gaussian process regressor can significantly boost the performance because of the temporal consistency measurement, from the results of "3-layers\GPR" shown in Fig. 3.

Comparison with Five State-of-the-art Methods

In order to prove the superiority of the proposed method, we compare our methods with five representative works, viz. off-the-shelf IForest (Liu, Ting, and Zhou 2012), One-class-SVM (OC-SVM) (Tax and Duin 2004), robust deep autoencoder (RDA) (Zhou and Paffenroth 2017), discriminative video anomaly detection framework (DVAD) (Giorno, Bagnell, and Hebert 2016), and incremental driving anomaly detection (IGRLSS) (Yuan, Fang, and Wang 2018).

We perform iForest, OC-SVM and RDA on the $Conv2$ layer of VGG-F model (Chatfield et al. 2014) with the size of 27*27*64 for each frame to detect the spatial anomaly regions, and we resize the spatial anomaly map as the same size of original frame by bilinear interpolation. Then, we detect the temporal anomaly frames by iForest, OC-SVM and RDA with the same feature representation of this work. Therefore, for a video, iForest, OC-SVM and RDA are temporally performed on a matrix with the size of 100*576 and generate anomaly vector with the size of 100*1. Actually, this separated detection strategy can get higher performances by iForest, OC-SVM and RDA for frame-level and pixel-level evaluation than that of putting all local region features of frames together[2]. For DVAD, we use the implementations of (Giorno, Bagnell, and Hebert 2016) which utilized the feature of (Lu, Shi, and Jia 2013). The frame-level ROC, pixel-level ROC curves, AUC and RD values are demonstrated in Fig. 5, and Table. 1. Because of the space limitation, the qualitative evaluation for temporal detection is provided in the supplemental files.

Table 1: The performance comparison (%) of distinct approaches.

Method	frame-AUCs	pixel-AUCs	RD
iForest	69.79	65.08	59.96
OC-SVM	55.01	67.77	62.18
iGRLSS	-	57.42	54.99
DVAD	53.04	68.50	63.60
RDA	62.28	69.41	65.55
Ours	**76.82**	**73.28**	**70.31**

These results clearly show that the proposed method ('Ours') is superior to others. DVAD demonstrates the worst frame-level performance, which is based on the assumption that the anomaly frames will not exceed 20% of the total number of frames of a video. However, over 50% of the videos violate this assumption. In addition, the dynamic camera motion makes DVAD detect anomaly regions in almost all frames of the dataset, which causes many false-alarms, as shown in Fig. 6(e). IGRLSS is an incremental method which assumes that the first ten frames of a video be normal, which is easily violated, thus the worst pixel-level anomaly is generated. On the contrary, the traditional IForest seems to be robust, and our method boosts its performance with 7.03% and 8.2% for frame-level and pixel-level detection, respectively. RDA is a deep autocoder which is also based on the rarity assumption of the anomaly in a video. Therefore, it may cause a wrong decision when the anomaly proportion is large. For example, Fig. 6(f) demonstrates adverse detection results.

Discussion on Efficiency

In this work, the major time cost is the feature extraction stage, which is nearly the same as the other methods. Therefore, we will compare the time cost for inferring the driving motion anomaly. Through observation, much timeis almost consumed by the temporal isolation forest decision layer. Based on the analysis in (Liu, Ting, and Zhou 2012), the time complexity of isolation forest construction is $\mathcal{O}(n\lambda \log s)$. We set $\lambda = 100$, $\beta = 256$ for each layer. The largest n in the first layer is $106*100 = 10600$ and decreases rapidly in the following layers. The normal motion sifting process (normal encoding) costs about 15 seconds for all the videos in this work, and abnormal motion sifting (abnormal decoding) spends about 6 seconds for each video on a PC platform with a 2.70GHz i7 CPU and 8GB RAM. Therefore, the proposed method runs fastest compared with other five methods. In addition, we also compare our method with others for the time cost per frame with the same PC platform, and the results are shown in Table. 2. From this table,

[2]We performed iForest, OC-SVM and RDA on the $Conv5$ layer of VGG-F model with the size of 13*13*256. For a video with 100 frames, we detected anomaly in a matrix of 16900 (13*13*100)*256, and obtained an anomaly score vector with size of 16900*1. We reshaped it back as 100*13*13 and enlarged the resolution as the original size of frame. We found iForest, OC-SVM and RDA determined all frames as anomaly in this strategy.

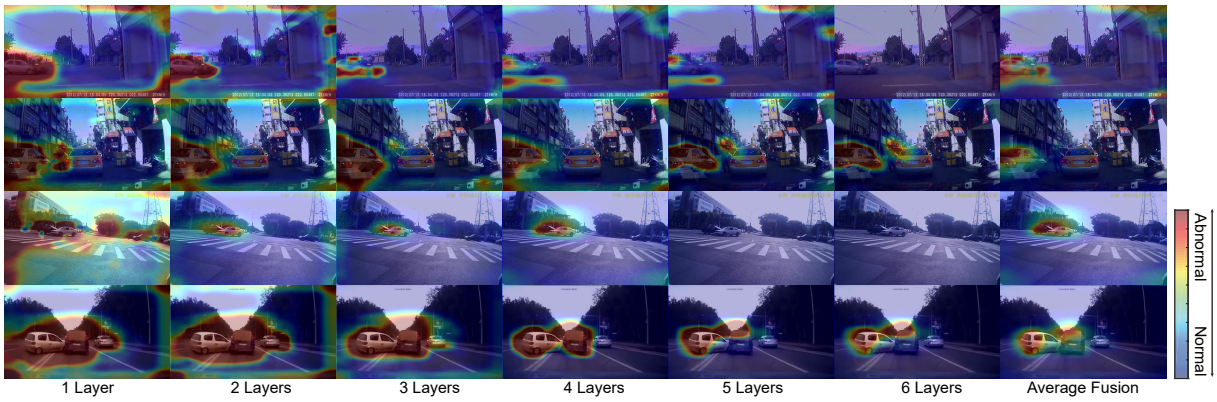

Figure 4: Some examples of detected spatial-anomaly maps when configuring different number of layers.

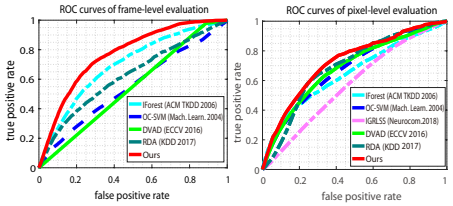

Figure 5: ROC curves of frame-level and pixel-level evaluation of various methods.

DVAD demonstrates the second efficiency, whereas RDA is very time consuming.

Table 2: The running time (secs/frame) comparison between the proposed method ('Ours') with five approaches.

Method	iForest	OC-SVM	iGRLSS	DVAD	RDA	Ours
Time cost	3	8	1.5	0.6	56	**0.06**

Conclusions

This work addressed the driving motion anomaly detection problem by an unsupervised multi-layer sifting process, which is fulfilled by a simple but effective Multi-layer Isolation-based Sifter (MLayer-iSifter) constructed by alternatively spatial-temporal *pruning* layers and *isolation forest decision* layers. By learning the normal motion representation with group videos, this framework provided a compact but discriminative normal motion representation for determining the driving motion anomaly, and can efficiently and simultaneously detect the temporal anomaly frames and spatial anomaly regions. Extensive experimental results on a dataset with 106 videos manually labeled carefully by ourselves demonstrate that the proposed method outperforms five state-of-the-art approaches.

References

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

Chan, F.; Chen, Y.; Xiang, Y.; and Sun, M. 2016. Anticipating accidents in dashcam videos. In *Asian Conference on Computer Vision*, 136–153.

Chandola, V.; Banerjee, A.; and Kumar, V. 2009. Anomaly detection:a survey. *ACM Computing Surveys* 41(3):1–58.

Chatfield, K.; Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Return of the devil in the details: Delving deep into convolutional nets. In *British Machine Vision Conference*.

Chen, Y.; Qian, J.; and Saligrama, V. 2013. A new one-class SVM for anomaly detection. In *International Conference on Acoustics, Speech and Signal Processing*, 3567–3571.

Cheng, K. W.; Chen, Y. T.; and Fang, W. H. 2015. Gaussian process regression-based video anomaly detection and localization with hierarchical feature representation. *IEEE Transactions on Image Processing* 24(12):5288–5301.

Giorno, A. D.; Bagnell, J. A.; and Hebert, M. 2016. A discriminative framework for anomaly detection in large videos. In *European Conference on Computer Vision*, 334–349.

Jiang, F.; Wu, Y.; and Katsaggelos, A. K. 2009. A dynamic hierarchical clustering method for trajectory-based unusual video event detection. *IEEE Transactions on Image Processing* 18(4):907–913.

Kataoka, H.; Suzuki, T.; Aoki, Y.; and Satoh, Y. 2018a. Anticipating traffic accidents with adaptive loss and large-scale incident DB. In *IEEE Conference on Computer Vision and Pattern Recognition*, 54–60.

Kataoka, H.; Suzuki, T.; Aoki, Y.; and Satoh, Y. 2018b. Drive video analysis for the detection of traffic near-miss incidents. In *IEEE International Conference on Robotics and Automation*, 54–60.

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

Kiran, B. R.; Thomas, D. M.; and Parakkal, R. 2018. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. *CoRR* abs/1801.03149.

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

Laxhammar, R., and Falkman, G. 2014. Online learning and sequential anomaly detection in trajectories. *IEEE Transactions on Patten Analysis and Machine Intelligence* 36(6):1158–1173.

Li, W.; Mahadevan, V.; and Vasconcelos, N. 2014. Anomaly detection and localization in crowded scenes. *IEEE Transactions on Patten Analysis and Machine Intelligence* 36(1):18–32.

Liu, W.; Luo, W.; Lian, D.; and Gao, S. 2018. Future frame prediction for anomaly detection - A new baseline. 3313–3320.

Liu, F. T.; Ting, K. M.; and Zhou, Z. H. 2012. Isolation-based

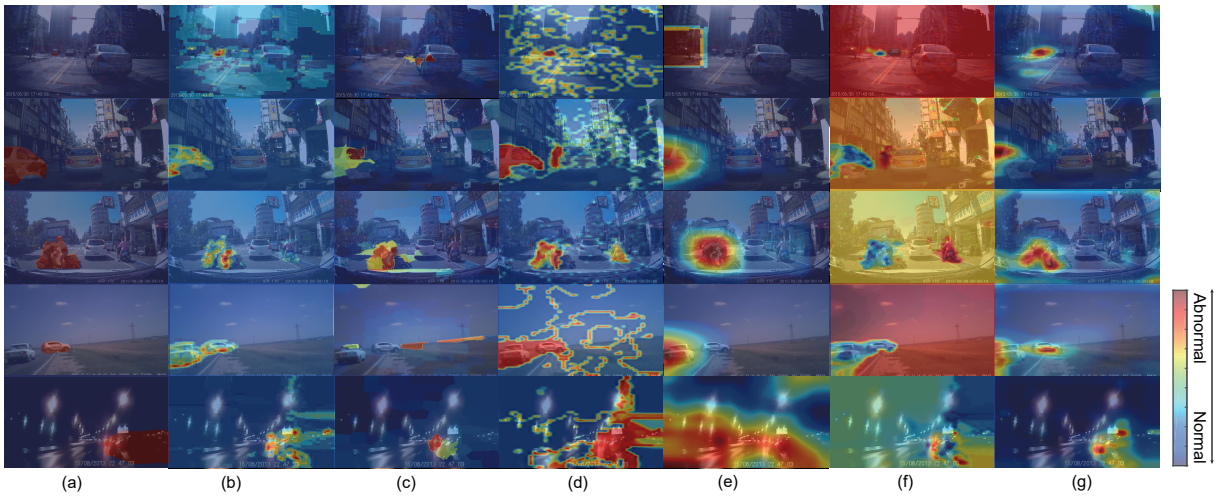

Figure 6: Some examples of spatial-anomaly regions w.r.t., different methods. (a) is the original image covered by the ground-truth; (b), (c), (d), (e), (f), and (g) are the detection results identified by pseudo color by IForest (Liu, Ting, and Zhou 2012), OC-SVM (Tax and Duin 2004), IGRLSS (Yuan, Fang, and Wang 2018), DVAD (Giorno, Bagnell, and Hebert 2016), RDA (Zhou and Paffenroth 2017) and Ours, respectively.

anomaly detection. *ACM Transactions on Knowledge Discovery from Data* 6(1):1–39.

Liu, C. 2009. *Beyond pixels: exploring new representations and applications for motion analysis*. Massachusetts Institute of Technology.

Lu, C.; Shi, J.; and Jia, J. 2013. Abnormal event detection at 150 FPS in MATLAB. In *IEEE International Conference on Computer Vision*, 2720–2727.

Luo, W.; Liu, W.; and Gao, S. 2017. A revisit of sparse coding based anomaly detection in stacked RNN framework. In *IEEE International Conference on Computer Vision*, 341–349.

Mehran, R.; Oyama, A.; and Shah, M. 2009. Abnormal crowd behavior detection using social force model. In *IEEE Conference on Computer Vision and Pattern Recognition*, 935–942.

Ounpraseuth, S. T. 2006. *Gaussian Processes for Machine Learning*. MIT Press.

Rao, A. S.; Gubbi, J.; Marusic, S.; and Palaniswami, M. 2016. Crowd event detection on optical flow manifolds. *IEEE Transactions on Cybernetics* 46(7):1524–1537.

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

Sabokrou, M.; Fayyaz, M.; Fathy, M.; and Klette, R. 2017. Deep-cascade: Cascading 3D deep neural networks for fast anomaly detection and localization in crowded scenes. *IEEE Transactions on Image Processing* 26(4):1992–2004.

Tax, D. M. J., and Duin, R. P. W. 2004. Support vector data description. *Machine Learning* 54(1):45–66.

Tran, H. T., and Hogg, D. 2017. Anomaly detection using a convolutional winner-take-all autoencoder. In *British Machine Vision Conference*.

Vedaldi, A., and Lenc, K. 2015. Matconvnet: Convolutional neural networks for MATLAB. In *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference, MM '15, Brisbane, Australia, October 26 - 30, 2015*, 689–692.

Xu, J.; Shi, C.; Qi, C.; Wang, C.; and Xiao, B. 2018. Unsupervised part-based weighting aggregation of deep convolutional features for image retrieval. In *AAAI Conference on Artificial Intelligence*.

Yu, F.; Wang, D.; and Darrell, T. 2018. Deep layer aggregation. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

Yuan, Y.; Fang, J.; and Wang, Q. 2015. Online anomaly detection in crowd scenes via structure analysis. *IEEE Transactions on Cybernetics* 45(3):548–561.

Yuan, Y.; Fang, J.; and Wang, Q. 2018. Incrementally perceiving hazards in driving. *Neurocomputing* 282:202–217.

Zhang, M.; Chen, C.; Wo, T.; Xie, T.; Bhuiyan, M. Z. A.; and Lin, X. 2017. Safedrive: Online driving anomaly detection from large-scale vehicle data. *IEEE Transactions on Industrial Informatics* 13(4):2087–2096.

Zhao, B.; Li, F.; and Xing, E. P. 2011. Online detection of unusual events in videos via dynamic sparse coding. In *IEEE Conference on Computer Vision and Pattern Recognition*, 3313–3320.

Zhou, C., and Paffenroth, R. C. 2017. Anomaly detection with robust deep autoencoders. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 665–674.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning deep features for discriminative localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2921–2929.