Spatio-Temporal AutoEncoder for Video Anomaly Detection

Yiru Zhao*
Shanghai Jiao Tong University
Alibaba Group
yiru.zhao@sjtu.edu.cn

Bing Deng
Alibaba Group
dengbing.db@alibaba-inc.com

Chen Shen†
Zhejiang University
Alibaba Group
zjushenchen@gmail.com

Yao Liu
Alibaba Group
xuanyao0111@gmail.com

Hongtao Lu‡
Shanghai Jiao Tong University
htlu@sjtu.edu.cn

Xian-Sheng Hua§
Alibaba Group
huaxiansheng@gmail.com

ABSTRACT

Anomalous events detection in real-world video scenes is a challenging problem due to the complexity of "anomaly" as well as the cluttered backgrounds, objects and motions in the scenes. Most existing methods use hand-crafted features in local spatial regions to identify anomalies. In this paper, we propose a novel model called Spatio-Temporal AutoEncoder (ST AutoEncoder or STAE), which utilizes deep neural networks to learn video representation automatically and extracts features from both spatial and temporal dimensions by performing 3-dimensional convolutions. In addition to the reconstruction loss used in existing typical autoencoders, we introduce a weight-decreasing prediction loss for generating future frames, which enhances the motion feature learning in videos. Since most anomaly detection datasets are restricted to appearance anomalies or unnatural motion anomalies, we collected a new challenging dataset comprising a set of real-world traffic surveillance videos. Several experiments are performed on both the public benchmarks and our traffic dataset, which show that our proposed method remarkably outperforms the state-of-the-art approaches.

CCS CONCEPTS

• **Computing methodologies** → **Computer vision**; *Machine learning approaches*;

KEYWORDS

Video Anomaly Detection, 3D Convolutions, AutoEncoder

1 INTRODUCTION

Automatic detection of abnormal events in video streams is a fundamental challenge in intelligent video surveillance systems, and has drawn increased attention from both academia and industry in the last a few years [2, 3, 12, 23, 32]. Differing from the supervised video analysis problems such as action recognition [5] and events detection [20], video anomaly detection is subject to two major difficulties: One is the data unbalance between positive and negative samples (that is, anomalous events, as positive samples, are much fewer than regular events). The other is the high variance within positive samples (anomalous events may contain a great variety of different cases though generally only limited training data is available). Due to the sparsity of positive samples, typical supervised event detection and recognition algorithms are not applicable for this task. This problem is generally addressed by training a model to represent regular activities in the video sequences and then regard the outliers as the abnormal events, which evidently diverge from the learned model.

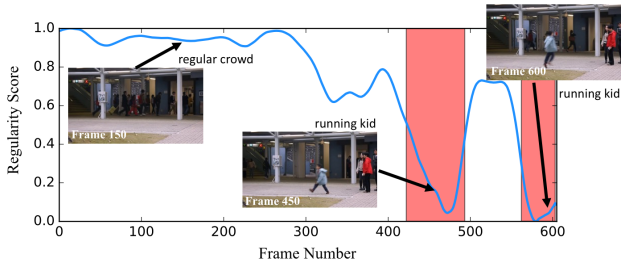

Figure 1: Regularity Score of a video sequence from CUHK Avenue dataset[12]. Red regions represent ground truth abnormal frames. The regularity score descents when anomalous event occurs.

Given the training data which generally only contains normal videos, learning the feature representation of regular activities is an unsupervised learning problem. A category of previous anomaly detection works [1–3, 16, 22] focus on modeling spatio-temporal event patterns of local 2D image patches or 3D video cubes by hand-crafted features extracted from low-level appearances and motions, e.g. histogram of oriented gradients(HOG), histogram of optical flow(HOF), 3D spatio-temporal gradient, etc. However, due to the limited representation capability of hand-crafted features, this category of previous approaches are not suitable for analyzing complex video surveillance scenes.

*This work was done when the author was visiting Alibaba as a research intern.
†This work was done when the author was visiting Alibaba as a research intern.
‡Corresponding author.
§Corresponding author.

MM'17, October 23–27, 2017, Mountain View, CA, USA.
 ISBN 978-1-4503-4906-2/17/10…$15.00
DOI: https://doi.org/10.1145/3123266.3123451

Deep learning methods have shown obvious advantages in feature learning and have been proven highly effective in solving discriminative vision tasks. Unsupervised deep learning approaches based on autoencoder networks have also been raised, as the second category of approaches, to address video anomaly detection problem [3, 32]. However, these methods merely rely on fully-connected autoencoder or 2d-convolutional autoencoder, without leveraging features from temporal dimensions, thus fail to capture the temporal cue of abnormal events, which is essential for identifying video event outliers.

Inspired by the superior performance of 3D convolutional networks in video analysis [5, 27], we propose a Spatio-Temporal(ST) AutoEncoder for video anomaly detection by applying 3D convolution in the encoder and 3D deconvolution in the decoder, which enhance the capability of extracting motion patterns from temporal dimensions. In addition to the reconstruction loss used in typical autoencoders, we also introduce a weight-decreasing prediction loss for predicting future frames, which guides the model to capture the trajectory of moving objects and enforces the encoder to extract the temporal features better. After training on normal video data, the autoencoder is supposed to reconstruct regular video clips with low error while high error will be incurred for reconstructing irregular clips. A regularity scores of each frame in the video sequence is then computed from the error and used to identify anomalous events, as shown in Figure 1.

Anomalous events in most real-world situations are highly complex, while most of current anomaly detection datasets only contains appearance anomalies or factitious motion anomalies. To evaluate the practicability of our proposed method, we collected a new challenging dataset comprising a set of real-world traffic surveillance videos about city traffic. The experiments show that our model is applicable to this complex application.

The main contributions of this paper can be summarized as follows:

- We propose a novel spatio-temporal autoencoder deep network, which is able to model regular video data from both spatial and temporal dimensions by performing 3D convolutions. To our best knowledge, this is the first 3D-convolution based video anomaly detection model.
- A weight-decreasing prediction loss is introduced in model training, which improves the performance of detecting anomalous events.
- We collect a new anomaly detection dataset consisting of real-world traffic surveillance videos, and show that our method outperforms the state-of-the-art approaches on both public benchmarks and our Traffic dataset.

2 RELATED WORK

We consider the anomaly detection and the 3D convolutional neural networks are two mostly related areas to our work. We also discuss the anomaly detection datasets in this section.

2.1 Anomaly Detection

Most video anomaly detection methods contain a local hand-crafted feature extracting step followed by a model training step. Outlier diverging from the trained model are regarded as anomalous events. Cong *et al.* [2] introduced the sparse reconstruction cost over the normal dictionary to measure the regularity of the testing sample based on Multi-scale Histogram of Optical Flow (MHOF). Mehran *et al.* [16] propose a social force model with optical flow features to detect abnormal behaviors in crowd scenes. Kartz *et al.* [8] presented a HMM-based approach to detect anomalies using 3D Gaussian distributions of spatio-temporal gradients. However, the hand-crafted features are not able to handle complex video surveillance scenes due to their limited representativeness

Deep learning methods have shown obvious advantages in feature learning and have been proven highly effective in various computer vision tasks, as well as anomaly detection. Xu *et al.* [32] propose a stacked autoencoder to automatically learn feature representation and utilize one-class SVM models to predict the anomaly scores. This method extracts local image patches and then flattens them to 1D vectors as the input of the fully-connected autoencoder, discarding the spatial information. The temporal information is provided by the optical flow patches, which merely contain the motion feature between two frames. Hasan *et al.* [3] propose a fully-convolutional autoencoder to learn spatio-temporal regularity. Although the proposed model takes multiple frames as input, temporal information is collapsed completely since the convolution operations are performed only spatially. In this paper we develop a new model by employing the 3D convolution operation to extract features from both spatial and temporal dimensions, which overcomes the disadvantages of existing deep learning based approaches.

2.2 3D Convolutional Neural Networks

As aforementioned, Deep Neural Networks (DNNs) have been proven superior in discriminative computer vision tasks, such as image classification [9], object detection [21], semantic segmentation [10], as well as a serious of video analysis problems [26, 30, 31, 35]. An early work [5] extends deep convolutional neural network to 3-dimensional for learning spatio-temporal features of videos. Tran *et al.* [27] train deep 3D convolutional networks on a large-scale video dataset [6] and achieve state-of-the-art performance on action recognition. Varol *et al.* [28] extend the 3D convolutional networks to long-term temporal convolution structures and achieve significant improvement over original models. The previous works have shown that 3D-ConvNets perform better than 2D-ConvNets for video analysis problems.

Although DNN-based approaches tackle the event detection and recognition problems effectively, they are impractical to be applied in video anomaly detection problem due to the sparsity of positive data samples. The anomalous video data is much fewer than the usual data because of the low probability of anomaly events. The large unbalance between positive and negative samples prevent us from training a discriminative model to differentiate anomaly from regular events. Meanwhile, the patterns of anomalous events have a high variety which is difficult to be covered by the limited training data. A typical binary event classifier will not work well due to the high variance within the abnormal class. Although supervised DNNs are unsuitable for anomaly detection, unsupervised DNNs based on autoencoder networks [3, 32] have been investigated to address this problem, but temporal dimension has not been

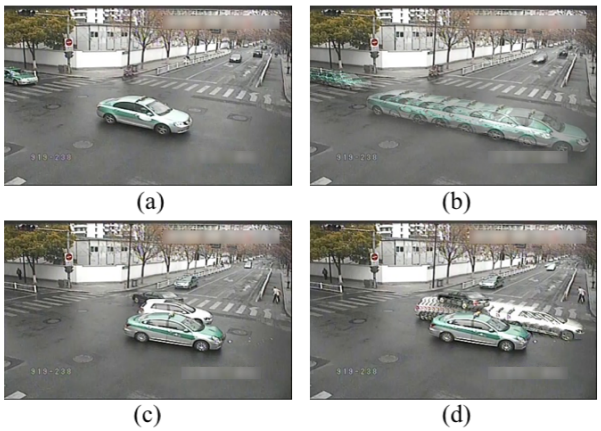

Figure 2: Two examples in our Traffic dataset. Top row: regular clip. Bottom row: irregular clip. Left column: single frame. Right column: multiple adjacent frames of the left frame fused in one image. It shows that the accident anomalies are global temporal events and can not be detected by focusing on single frame or local objects.

well leveraged. To overcome the disadvantages in existing unsupervised DNN based approaches for anomaly detection, inspired by the significant improvement involved by 3D-ConvNets in video analysis, we use 3D convolutional autoencoder to model regular video content, based on which we detect anomaly from complex surveillance videos. Besideds the adoption of 3D CNN into abnormality detection, we also propose a weight-decreasing prediction loss in model training stage, which improves the performance of detecting anomalous events.

2.3 Anomaly Detection Datasets

There have been several published anomaly detection datasets, while most of them only focus on appearance anomalies or factitious motion anomalies. The UCSD pedestrian dataset [14] provides two different crowded scenes and the anomalies include bicycles, skateboarders, vehicles and wheelchairs. All the anomalous objects have different appearance from the normal pedestrians in the training data thus can be detected from single frame without motion features. The UMN dataset consists of three different scenes of crowds of walking people who suddenly started running. The anomalies are unnaturally occurring, *i.e.* they are staged for the purposes of assembling the dataset. The algorithms evaluated on these existing datasets may be impractical to be applied in real-world applications.

Besides presenting the deep ST AutoEncoder network, we provide a real-world traffic surveillance dataset in this paper, in which real accidents are considered as the anomalous events. Here an anomalous clips is defined as a set of consecutive frames which record the accident. Traffic accident detection is a challenging problem in modern intelligent surveillance system [33], because the accident anomalies are global temporal events and can not be detected by checking single frame or local objects only. Taking Figure 2 as an example, the regular clip (top row) displays a vehicle passing through the crossroads, and the irregular one captures two vehicles crashed and stayed in the scene. The regular clip and irregular clip can not be distinguished by a single frame. Even given multiple continuous frames, the motion states of each vehicle are regular, *i.e.* moving forward or staying behind are both normal motion states of single vehicle. The anomaly lies in the irregular traffic flow, which often appears as different motion states of vehicles in the crossroads, anomalous trajectory, as well as traffic jam caused by the accidents. This kind of global spatio-temporal anomalies increase the difficulty of our Traffic dataset.

3 OUR METHOD

For concreteness, we first briefly review the 3D convolution and then discuss the details of our proposed model.

3.1 3D Convolution

The typical 2D-ConvNets apply convolutions on the 2D feature maps to extract features from the spatial dimensions only. Denote the value of cell in the i^{th} channel at position (x, y) in the l^{th} layer as α_l^{ixy}, it is calculated by:

$$\alpha_l^{ixy} = f\Big(\sum_{k=1}^{K_l}\sum_{h=1}^{H_l}\sum_{w=1}^{W_l}\theta_{l,i}^{khw}\alpha_{l-1}^{k(x+h)(y+w)} + \beta_l^i\Big) \tag{1}$$

where $\theta_{l,i}^{khw}$ denotes the value of cell at position (h, w) of k^{th} channel in the i^{th} kernel connected to the l^{th} layer. H_l, W_l and K_l are height, width and channel number of the kernel, and K_l equals the kernel number in the $(l-1)^{th}$ layer. β_l^i is the bias for the i^{th} feature map in the l^{th} layer. $f(\cdot)$ is the activation function, which are typically nonlinear transformations such as sigmoid, tanh or ReLU [17].

2D-ConvNets have shown superior performance in image recognition, while they are incapable of capturing the temporal information encoded in consecutive frames for video analysis problems. Ji *et al.* [5] propose to perform 3D convolutions to compute features from both temporal and spatial dimensions by convolving a 3D kernel to the cubic formed by catenating multiple contiguous frames in the temporal dimension. Thus the value of the cell in the i^{th} channel at position (x, y, z) in the l^{th} layer is calculated by:

$$\alpha_l^{ixyz} = f\Big(\sum_{k=1}^{K_l}\sum_{h=1}^{H_l}\sum_{w=1}^{W_l}\sum_{d=1}^{D_l}\theta_{l,i}^{khwd}\alpha_{l-1}^{k(x+h)(y+w)(z+d)} + \beta_l^i\Big) \tag{2}$$

where D_l is the depth of 3D kernel. The feature maps are connected to D_l contiguous frames in the previous layer to capture motion information.

Based on the 3D convolution described above, we will introduce the Spatio-Temporal AutoEncoder (ST AutoEncoder) that we developed for video anomaly detection in the following sections.

3.2 3D Convolutional AutoEncoder

Input Data. In most typically CNNs for image recognition, the input data is a single image with three channels (*e.g.* R, G and B color channels). While in anomaly detection networks, the input data is a video clip consisting of multiple frames. Hasan *et al.* [3] construct the input by a temporal cuboid using a sliding window.

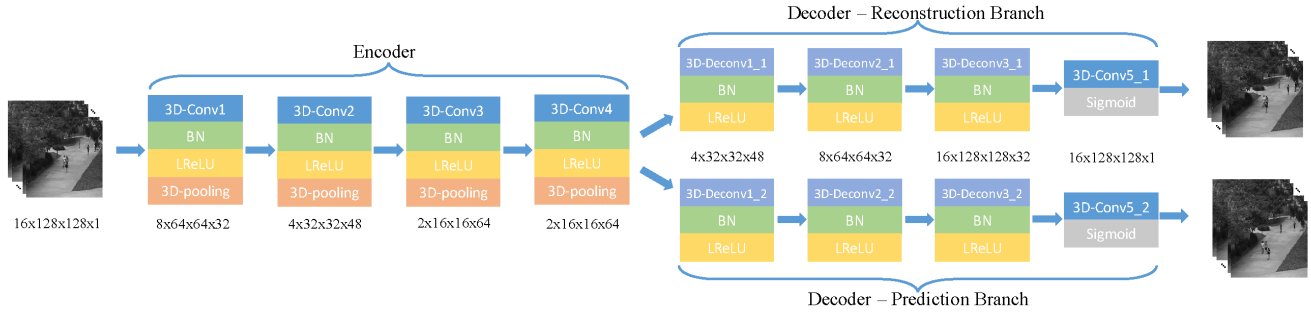

Figure 3: Architecture of the network. An encoder followed by two branches of decoder for reconstructing past frames and predicting future frames respectively.

Specifically, this method stacks T frames in the channel dimension and input them to the autoencoder, where T is the length of the sliding window. However, the temporal information is collapsed completely after the first convolution layer, because the 2D convolution operations are performed only spatially. The T channels (or $3T$ channels for RGB frames) input data are connected jointly to each channel of the first feature map, so that the temporal features are rarely preserved. To solve this problem, we construct the input as a hyper-cuboid by stacking T frames in the 4^{th} dimension (usually called temporal dimension) and perform 3D convolutions on it. The channel number is equal to the original frames and depends on the data type, *e.g.* 1 for grayscale frames, 3 for RGB frames, 2 for optical flow (optflow-x and optflow-y). All elements of in input data are normalized to $[0, 1]$.

Data Augmentation. Due to the large amount of parameters in DNNs, large scale data is needed to train the model generally. However, the size of training data in anomaly detection datasets typically is not sufficiently large enough, comparing to the events recognition datasets, *e.g.* UCF101 [24], Sports-1M [6]. Therefore, we generate more input hyper-cuboids with various transformations (random cropping, brightness changing and Gaussian blurring) applied on the clips sampled from the video sequences. Note that the method in [3] performs data augmentation in the temporal dimension by sampling frames with stride-1, stride-2 and stride-3. Stride-x sampling increases the moving speed of the objects in the input sequence by x times. However, the speed is an important temporal feature in many anomaly detection scenarios, therefore we use constant stride to sample frames in our method, so that the moving speed of objects remains unchanged.

Network Architecture. Our proposed ST AutoEncoder network is illustrated in Figure 3. We set the frame number $T = 16$, resize each frame to 128×128 and use grayscale image with 1 channel, thus the shape of input hyper-cuboid is $16 \times 128 \times 128 \times 1$. The encoder contains four 3D convolutional layers to extract the spatio-temporal features from the input video clip. Tran *et al.* [27] find that $3 \times 3 \times 3$ convolution kernel with stride $1 \times 1 \times 1$ for all layers works best in action recognition tasks, hence we use the same kernel size and stride for all 3D convolutional layers in our network. The output feature map of each kernel is a 3D tensor with temporal dimension (instead of a 2D matrix in 2D-ConvNets, in which lose temporal information of the input signal is lost right after every convolutional operation), as 3D-ConvNets preserve the temporal information resulting in the output volume. We increase the number of kernels to extract complex semantic features in deep layers. Batch normalization [4] is applied to each of the convolutional layers, which accelerates convergence at the training stage. Leaky ReLU [13] is used after batch normalization. We use 3D max-pooling layers with kernel size $2 \times 2 \times 2$ and stride $2 \times 2 \times 2$ after each 3D convolutional layers except the last one. After the encoder part, the bottleneck hidden layer of the autoencoder has shape of $2 \times 16 \times 16 \times 64$ and contains the spatial-temporal features encoded from the input video clips. The decoder part has a symmetric structure with respect to the encoder part. We stack three 3D deconvolutional layers followed by one 3D convolutional layer as the reconstruction branch, to rebuild the input signal from the hidden layer. The deconvolution operation is proposed in [34] and is actually the gradient operation of convolution. We use 3D deconvolution layers with stride $2 \times 2 \times 2$ without unpooling layer in the decoder part. Sigmoid is applied to the last layer to cater to the normalized input data. The proposed ST AutoEncoder network has another branch in the decoder part besides the reconstruction branch and we will describe it in detail in Section 3.3.

Similar to previous works [3, 32], the reconstruction error is expressed as the Euclidean loss:

$$L_{rec} = \frac{1}{N} \sum_{i=1}^{N} \|X_i - f_{rec}(X_i)\|_2^2 \tag{3}$$

where X_i is the i^{th} hyper-cuboid of the input batch of size N, and $f_{rec}(X_i)$ is the output of the reconstruction branch.

3.3 Weight-decreasing Prediction Loss

Inspired by the previous works [11, 15, 18, 25, 29] which prove that predictive networks benefit learning video representations, we design a prediction branch in the decoder part to predict the future T frames after the input video clip. Specifically, the reconstruction branch and the prediction branch share the same hidden feature layer but perform different tasks: reconstructing the past sequence and predicting the future sequence respectively. The prediction task guides the model to capture the trajectory of moving objects and enforce the encoder to better extract the temporal features.

The prediction loss function used in previous works [25, 29] is Euclidean loss, which is same with the reconstruction loss in Equation 3. Each frame of the predicted video clip has the same weight to the overall loss. However, the constant-weight prediction loss is unsuitable for anomaly detection tasks. In most scenarios of video anomaly detection, the viewpoint is fixed and various objects go in and out. The appearance of new objects is hard to predict, effecting the convergence of the prediction network at the training stage. We apply the prediction loss to enforce the model to extract the motion feature of the existing objects and predict their movements in near future, rather than predicting the appearance of new objects in relatively distant future. The probability of the appearance of new objects gradually increases as time goes on, hence we impose a decreasing weight on each frame of the predicted video clip. The prediction loss is formulated by:

$$L_{pred} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{T^2}\sum_{t=1}^{T}(T-t)\|X_{i+T}^{t} - f_{pred}(X_i)^t\|_2^2 \quad (4)$$

where X_i is the input hyper-cuboid, $f_{pred}(X_i)$ is the output of the prediction branch, X_{i+T} denotes the ground truth of the future T frames and the superscript t in X^t denotes the t^{th} frame of the video clip X. The t^{th} frame has a weight of $T-t$, which decreases as t increases.

3.4 Regularity Score

With the reconstruction loss and prediction loss, the optimization objective of the proposed model is defined by:

$$\min_{W} \; L_{rec} + L_{pred} + \lambda\|W\|_2^2 \quad (5)$$

where W is the parameters of the model, λ is a regularization parameter that controls the model complexity.

Once the model is trained, the reconstruction error of the test video sequence x is given by $e(x) = L_{rec}(x)$, then we normalize the reconstruction error of the sequences from the same video to calculate the regularity score as follows:

$$s(x) = 1 - \frac{e(x) - min_x e(x)}{max_x e(x)} \quad (6)$$

Video sequences consisting of regular events have a higher regularity score as they are close to the normal training data in the feature space. On the contrary, the anomalous sequences have a lower regularity score, thus it can be used to locate anomalies. In particular, it is impractical to calculate $min_x e(x)$ and $max_x e(x)$ in the real-time anomaly detection system because the future data is unobservable. These two value should be set experimentally according to the historical data.

4 EXPERIMENTS

4.1 Datasets

We evaluated the proposed ST AutoEncoder in three datasets, including two existing datasets, UCSD Pedestrian [14] and CUHK Avenue [12], and the newly collected Traffic dataset as aforementioned.

UCSD Pedestrian has two different scenes, Ped1 and Ped2. Ped1 has 34 short clips for training and another 36 clips for testing. Each clip has 200 frames with a resolution of 238 × 158. Ped2 has 16 short clips for training and another 12 clips for testing. Each clip has about 170 frames with a resolution of 360 × 240. The anomalies include bicycles, skateboarders, wheelchairs and vehicles within the regular crowd.

Table 1: Comparing anomaly detection datasets. 'Nor' for normal frames and 'Abn' for abnormal frames

Dataset	#Scene	#Train	#Test Nor	#Test Abn
UCSD Pedestrian	2	9350	3569	5641
CUHK Avenue	1	15328	11612	3712
Traffic	5	248543	19784	59562

CUHK Avenue has 16 training and 21 testing video clips. Each clip is about 1 minutes long with a resolution of 640 × 360. The anomalies include some unusual action of people in the clips, *e.g.* running, loitering, or throwing.

Traffic dataset is collected from a real-world traffic surveillance system and contains 5 different scenarios. For each scenario, we provide a normal video of about 30 minutes to train the model and a testing video of about 10 minutes. An accident occurs in the testing video and the related and subsequent frames are considered as anomalous because the accident vehicles stay in the shot and the traffic flow is influenced. The anomaly lies in the irregular traffic flow, which often appears as different motion states of vehicles in the crossroads, irregular trajectory, as well as traffic jam cased by the accidents. This kind of anomaly detection is a challenging problem in modern intelligent surveillance system and the Traffic dataset is able to evaluate the performance of anomaly detection algorithms applicable to real-world applications. [1]

The comparison of frame numbers of these three datasets is shown in Table 1. The newly collected Traffic dataset has the most scenarios and frame numbers. The complexity of the scenarios, variances of anomalies and the scale of the data make this benchmark much more challenging than existing datasets.

4.2 Anomalous Visualization

Once we trained the model, the regularity score can be calculated based on the reconstruction error. Video sequences consisting of regular events have lower error and the anomalous sequences have higher. The reconstruction error is calculated from every pixel in every frames, which enable us to split the error into each frame and locate the anomalous region in the picture.

Five groups of examples from different datasets are shown in Figure 4. In each group, the left column shows a sample frame from an anomalous event. The mid column displays the corresponding reconstruction output of our STAE model. The reconstruction error map is shown in the right column at pixel level. Blue represents low error and red represents high.

In the top three groups, each anomaly lies in only single object (vehicle, bicycler or running girl) and can be clearly located in the error map. The normal training data do not contain these abnormal objects, so that the model fail to reconstruct them, resulting in a higher error in the residual map. While in the bottom two groups of traffic scenarios, the anomalies lie in the irregular traffic flow

[1] The dataset will be released to public with the publication of this manuscript.

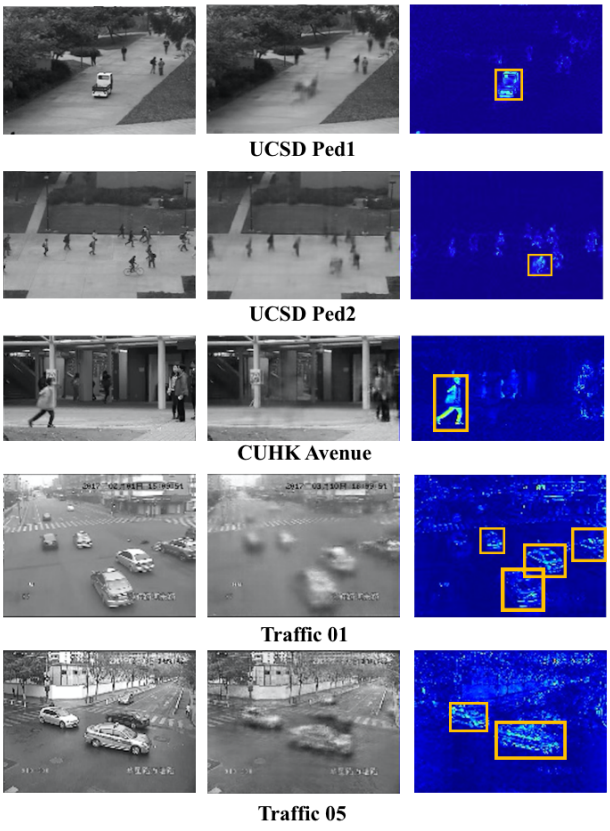

Figure 4: Anomalous visualization. Left column: frames from irregular clips. Mid column: the reconstruction output of our model. Right column: the reconstruction error map. The orange rectangles highlight the anomalous regions in the error maps. The anomalies lies in only single object in the top three scenes, while are associated with multiple objects in the bottom two.

instead of single vehicle. Specifically, the anomalies are indicated by that the accident vehicles stop in the crossroads while other vehicles pass by them. Therefore, the anomalous regions in the error maps are associated with multiple objects instead single object in the top three scenario. This phenomenon evidences that the traffic accidents are global spatio-temporal anomalies and the temporal dimension needs to be considered in the detection algorithm. We will display and discuss the quantitative results of our method with several state-of-the-art approaches on these three datasets in Section 4.3.

4.3 Anomaly Events Detection

Based on the reconstruction error, the regularity score is calculated by Equation 6, and can be further used to detect anomalous events. As shown in Figure 5, the regularity score of a video clip descents when anomaly occurs. The red regions denote the ground truth period of anomalous events. To evaluate our method quantitatively,

Table 2: Comparison on UCSD Pedestrian and CUHK Avenue datasets.

	Ped1		Ped2		Avenue	
Algorithm	AUC	EER	AUC	EER	AUC	EER
MPPCA[7]	59.0	40.0	69.3	30.0	-	-
SF[16]	67.5	31.0	55.6	42.0	-	-
SF+MPPCA[14]	66.8	32.0	61.3	36.0	-	-
MDT[14]	81.8	25.0	82.9	25.0	-	-
SCL[12]	91.8	**15.0**	-	-	-	-
AMDN[32]	92.1	16.0	90.8	17.0	-	-
ConvAE[3]	81.0	27.9	90.0	21.7	70.2	25.1
STAE-grayscale	**92.3**	15.3	**91.2**	**16.7**	77.1	33.8
STAE-optflow	87.1	18.3	88.6	20.9	**80.9**	**24.4**

we follow the evaluation metric in previous works [3, 7, 14]. The ROC curve is produced by varying the threshold and calculating the True Positive Rate(TPR) and the False Positive Rate(FPR). Then the anomaly detection algorithms are compared in terms of Area Under Curve (AUC) and Equal Error Rate (EER).

The comparison of our method with several state-of-the-art approaches on UCSD Pedestrian and CUHK Avenue datasets is shown in Table 2. To the best of our knowledge, there are no AUC/EER results for Avenue dataset besides ConvAE [3]. We consider two types of our STAE model based on different types of input data. STAE-grayscale takes the raw grayscale pixels as input and outperforms most of the state-of-the-art methods, expect that SCL [12] is slightly better than STAE-grayscale in terms of EER(15.0 vs. 15.3). Some of the methods use 3D gradient feature or Histograms of Optical Flows (HOF) to extract the temporal features, while STAE-grayscale model captures the motion feature from the raw video data automatically. STAE-optflow takes the optical flow calculated by the algorithm described in [19] as input, the optical flow data provides discriminative temporal features and improves the performance of the STAE model on CUHK Avenue dataset. The optical flow input is not able to improve the results on UCSD Pedestrian dataset because the outlier in this dataset are rather appearance anomalies while the appearance features are not contained by the optical flow. While for Avenue dataset, STAE-optflow improved more compared with STAE-grayscale by 4.9% in AUC. The results also demonstrate that our ST AutoEncoder model is applicable for different types of input data.

The same evaluation is performed on the newly collected Traffic dataset. We set ConvAE[3] as the state-of-the-art method because it has certain capability of revealing temporal features. The results of 5 scenarios are shown in Table 3, and the average results are also reported. All the tested models take grayscale frames as input. Although the optical flow input improves the results on Avenue datasets, we do not adopt it to the Traffic dataset because stopped vehicles are unable to provide optical flow information and the irregular traffic flow is difficult to be detected only by moving vehicles. We set a model called ConvAE-1frame as the base model, which is a 2D convolutional autoencoder and only takes one frame as input, achieving a baseline result without the temporal features.

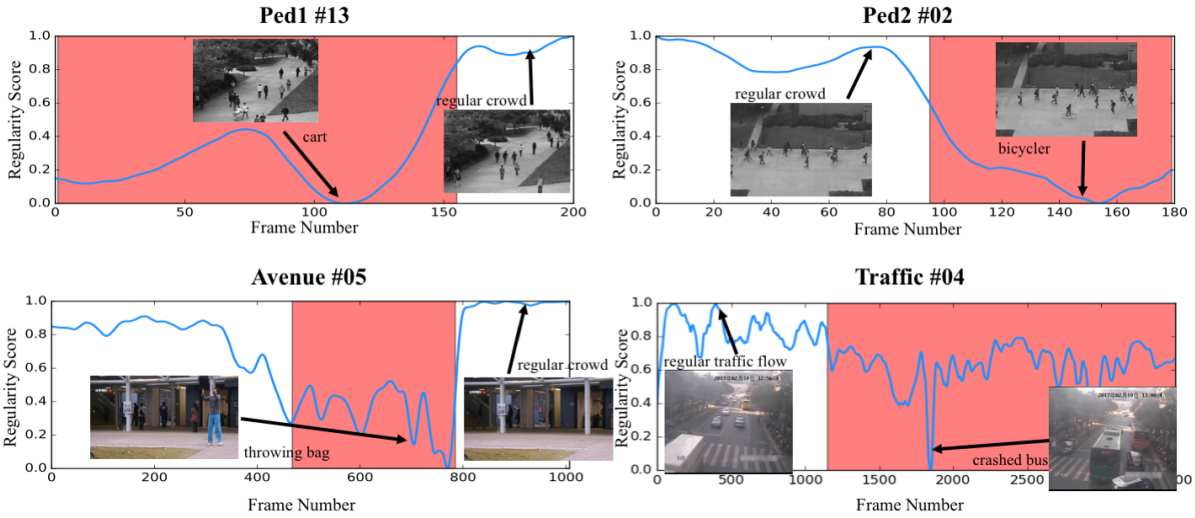

Figure 5: Regularity score curves of four testing video clips from the three datasets. Red regions represent ground truth anomalous frames. It shows that the regularity scores descent when anomalies occur. Several frames are sampled to display the regular/irregular events in each scenario. Best viewed in color.

Table 3: Comparison on Traffic dataset.

	#01		#02		#03		#04		#05		Average	
Algorithm	AUC	EER	AUC	EER	AUC	EER	AUC	EER	AUC	EER	AUC	EER
ConvAE-1frame	54.9	43.3	59.3	44.1	68.7	34.7	66.1	40.1	84.1	22.7	66.6	37.0
ConvAE[3]	57.2	48.6	75.7	28.7	71.2	31.2	71.0	38.3	82.8	24.2	71.5	34.2
STAE-3d-w/o-pred	71.4	36.4	78.7	29.8	77.4	28.3	75.6	34.9	85.8	19.6	77.8	29.8
STAE-3d-constant-pred	73.2	34.8	80.0	27.2	**77.6**	27.7	77.8	29.2	86.1	**19.0**	78.9	27.6
STAE-3d-decreasing-pred	**74.3**	**33.3**	**81.4**	**25.5**	76.6	**27.1**	**79.3**	**26.1**	**86.5**	19.1	**79.6**	**26.2**

The other four models take a video clip of $T = 16$ frames as input, and we sample the frames at stride-5 to ensure that the input clip overlays enough time period. With the multiple frames input, ConvAE increases the average AUC by 7.4% and decreases EER by 7.7%. However, the 2D-ConvNets in ConvAE fail to extract the temporal features effectively. To demonstrate the improvements of our proposed model, we report the results of three 3D-ConvNets models trained with different losses: STAE-3d-w/o-pred with reconstruction loss only, STAE-3d-constant-pred with an additional constant weight prediction loss, and STAE-3d-decreasing-pred with an additional decreasing weight prediction loss. STAE-3d-w/o-pred benefits from the 3D-ConvNets and outperforms ConvAE by 8.8% in AUC. The prediction loss enhances the video representation learning, thus improves the results further. Within the two models trained with both reconstruction and prediction loss, STAE-3d-decreasing-pred model performs better than STAE-3d-constant-pred model. Because the constant-weight prediction loss has the same weight on each frame and the model training is more likely to be influenced by the newly appearing objects in the later frames, so that we introduce the weight-decreasing prediction loss. Overall, our proposed method, denoted by STAE-3d-decreasing-pred, achieves the best result(increases AUC by 11.3% and decreases EER by 23.4% compared to ConvAE).

4.4 Predicting Future Frames

As aforementioned, we design a prediction branch in the ST AutoEncoder network in order to enhance the video representation learning by tracking the trajectories of moving objects in the video sequence.

Two examples are illustrated in Figure 6. Our STAE model is able to reconstruct the input regular video clips, as well as predicting the future frames. The trajectories of moving vehicles (marked in green) are well predicted in the future frames. We also give an example that there is a new vehicle (marked in red) entering the scene, showing that our model fails to predict this new vehicle. However, predicting the appearance of new objects is an open question and is not in consideration in the anomaly detection problem. The prediction loss is applied in order to enforce the model to extract the motion feature of the existing objects and predict their movements in near future, rather than predicting the appearance of new objects in relatively distant future. So that we impose the decreasing weight with respect to the time step on the prediction loss. The experimental results in

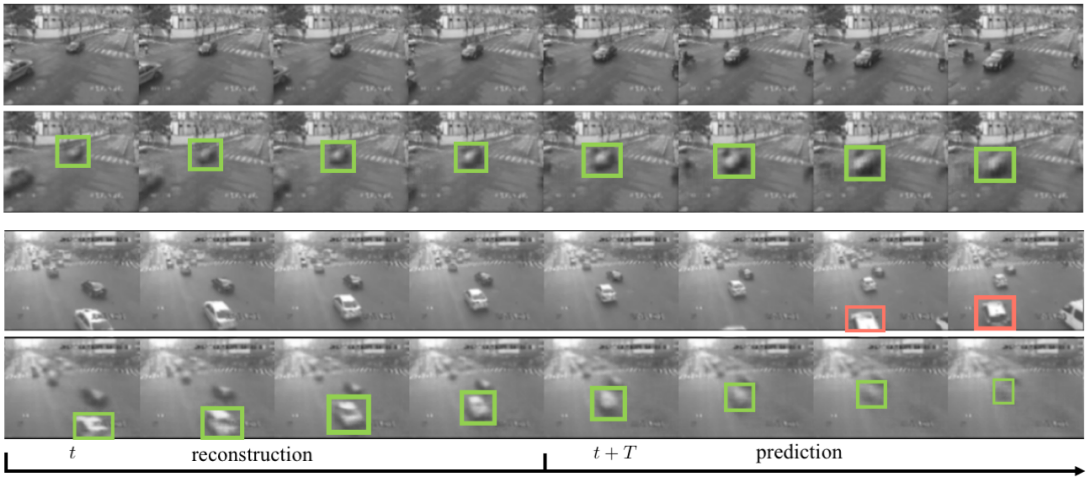

Figure 6: Two groups of frames prediction in Traffic dataset. The top row in each group is the ground truth video sequence and the bottom row is the reconstruction and prediction output of our network. The left ones are sampled from the T input frames and the right ones are from future clips. The moving vehicles are highlighted by green boxes. The new vehicle entering the scene are marked in red.

Section 4.3 demonstrate the superiority of the weight-decreasing prediction loss for video anomaly detection task.

5 CONCLUSION

We present a model called Spatio-Temporal AutoEncoder for video anomaly detection by utilizing 3D ConvNets to extract video features from both spatial and temporal dimensions, and in a multi-task (reconstruction and future prediction) manner. A weight-decreasing prediction loss is designed to enhance the motion representation learning. We also collected a new anomaly detection dataset from a real-world application. Qualitative analysis and quantitative comparison are performed on public benchmark and our Traffic dataset, and the results show that our proposed method outperforms state-of-the-art approaches.

Future works include investigating other network architectures, fusing multimodal input data(*e.g.* , RGB frame and optical flow), evaluating the regularity score at instance level instead of pixel level, and applying our framework to more complex scenarios.

ACKNOWLEDGMENTS

This paper is partially supported by NSFC (No.61272247, 61533012, 61472075), the 863 National High Technology Research and Development Program of China (SS2015AA020501), the Basic Research Project of "Innovation Action Plan" (16JC1402800) and the Major Basic Research Program (15JC1400103) of Shanghai Science and Technology Committee.

REFERENCES

[1] Yannick Benezeth, P-M Jodoin, Venkatesh Saligrama, and Christophe Rosenberger. 2009. Abnormal events detection based on spatio-temporal co-occurences. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.* IEEE, 2458–2465.

[2] Yang Cong, Junsong Yuan, and Ji Liu. 2011. Sparse reconstruction cost for abnormal event detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on.* IEEE, 3449–3456.

[3] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K Roy-Chowdhury, and Larry S Davis. 2016. Learning temporal regularity in video sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 733–742.

[4] Sergey Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15).* 448–456.

[5] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 2013. 3D convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence* 35, 1 (2013), 221–231.

[6] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition.* 1725–1732.

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

[8] Louis Kratz and Ko Nishino. 2009. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.* IEEE, 1446–1453.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems.* 1097–1105.

[10] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 3431–3440.

[11] William Lotter, Gabriel Kreiman, and David Cox. 2015. Unsupervised learning of visual structure using predictive generative networks. *arXiv preprint arXiv:1511.06380* (2015).

[12] Cewu Lu, Jianping Shi, and Jiaya Jia. 2013. Abnormal event detection at 150 fps in matlab. In *Proceedings of the IEEE International Conference on Computer Vision.* 2720–2727.

[13] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proc. ICML*, Vol. 30.

[14] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. 2010. Anomaly detection in crowded scenes. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE, 1975–1981.

[15] Michael Mathieu, Camille Couprie, and Yann LeCun. 2015. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440* (2015).

[16] Ramin Mehran, Alexis Oyama, and Mubarak Shah. 2009. Abnormal crowd behavior detection using social force model. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.* IEEE, 935–942.

[17] Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*. 807–814.

[18] Rasmus Berg Palm. 2012. Prediction as a candidate for learning deep hierarchical models of data. *Technical University of Denmark* 5 (2012).

[19] Javier Sánchez Pérez, Enric Meinhardt-Llopis, and Gabriele Facciolo. 2013. TV-L1 optical flow estimation. *Image Processing On Line* 2013 (2013), 137–150.

[20] Vignesh Ramanathan, Jonathan Huang, Sami Abu-El-Haija, Alexander Gorban, Kevin Murphy, and Li Fei-Fei. 2016. Detecting events and key actors in multi-person videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3043–3053.

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*. 91–99.

[22] Elisa Ricci, Gloria Zen, Nicu Sebe, and Stefano Messelodi. 2013. A prototype learning framework using emd: Application to complex scenes analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 3 (2013), 513–526.

[23] Venkatesh Saligrama and Zhu Chen. 2012. Video anomaly detection based on local statistical aggregates. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2112–2119.

[24] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* (2012).

[25] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. 2015. Unsupervised Learning of Video Representations using LSTMs.. In *ICML*. 843–852.

[26] Lin Sun, Kui Jia, Dit-Yan Yeung, and Bertram E Shi. 2015. Human action recognition using factorized spatio-temporal convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*. 4597–4605.

[27] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*. 4489–4497.

[28] Gül Varol, Ivan Laptev, and Cordelia Schmid. 2016. Long-term temporal convolutions for action recognition. *arXiv preprint arXiv:1604.04494* (2016).

[29] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. 2015. Anticipating the future by watching unlabeled video. *arXiv preprint arXiv:1504.08023* (2015).

[30] Limin Wang, Yu Qiao, and Xiaoou Tang. 2015. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4305–4314.

[31] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. 2016. Temporal segment networks: towards good practices for deep action recognition. In *European Conference on Computer Vision*. Springer, 20–36.

[32] Dan Xu, Elisa Ricci, Yan Yan, Jingkuan Song, and Nicu Sebe. 2015. Learning deep representations of appearance and motion for anomalous event detection. *arXiv preprint arXiv:1510.01553* (2015).

[33] Kimin Yun, Hawook Jeong, Kwang Moo Yi, Soo Wan Kim, and Jin Young Choi. 2014. Motion interaction field for accident detection in traffic surveillance video. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE, 3062–3067.

[34] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. 2010. Deconvolutional networks. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2528–2535.

[35] Wangjiang Zhu, Jie Hu, Gang Sun, Xudong Cao, and Yu Qiao. 2016. A key volume mining deep framework for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1991–1999.