Multi-Encoder Towards Effective Anomaly Detection in Videos

Zhiwen Fang, Joey Tianyi Zhou, Yang Xiao, Yanan Li, and Feng Yang

***Abstract*—Given normal training samples, anomaly detection in videos can be regarded as a challenging problem of identifying unexpected events. The state-of-the-art approaches generally resort to the autoencoder model by using a single encoder to capture the motion and content patterns jointly. Nevertheless, due to the lack of accurate labels of normal and abnormal samples, how to detect anomalies is decided by the subjective understanding of models. It infers that different models will prefer to mine different patterns according to the characteristics of models. We call this problem as a pattern bias problem. To alleviate this problem, a novel Multi-Encoder Single-Decoder network, termed as MESDnet, is proposed in the spirit of encoding motion and content cues individually with multiple encoders. MESDnet is of end-to-end learning ability and real-time running speed. Particularly, the differences between adjacent frames and the raw frames are used as the motion and content sources, respectively. Then, a decoder takes charge of detecting anomalies in the way of observing reconstructing error towards the video frames by using the multi-stream encoded motion and content features simultaneously. The experiments on the CUHK Avenue dataset, the UCSD Pedestrian dataset, and the ShanghaiTech Campus dataset verify the effectiveness of MESDnet.**

***Index Terms*—Anomaly detection, motion encoder, content encoder, multi-encoder single-decoder network.**

Manuscript received February 8, 2020; revised July 17, 2020 and September 10, 2020; accepted October 30, 2020. Date of publication November 18, 2020; date of current version November 18, 2021. This work was supported in part by the National Natural Science Foundation of China under Grants 61702182, 61771233, 61502187, and 61906139, in part by the Hunan Provincial Natural Science Foundation of China under Grant 2018JJ3254, in part by the Agency for Science, Technology and Research (A*STAR) under its AME Programmatic Funding Scheme Project A18A1b0045, in part by the National Key Laboratory Open Fund of China under Grant 6142113180211, and in part by the Hubei Provincial Natural Science Foundation of China under Grant 2019CFB173. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. A. D. Bagdanov. *(Corresponding author: Joey Tianyi Zhou.)*

Zhiwen Fang is with the School of Biomedical Engineering, Southern Medical University, Guangzhou 510515, China, and also with the School of Energy and Mechanical-electronic Engineering, Hunan University of Humanities, Science and Technology, Loudi 417000, China (e-mail: fzw310@gmail.com).

Joey Tianyi Zhou is with the A*STAR, IHPC, Singapore 138632, Singapore (e-mail: joey.tianyi.zhou@gmail.com).

Yang Xiao is with the National Key Laboratory of Science and Technology on Multispectral Information Processing, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: Yang_Xiao@hust.edu.cn).

Yanan Li is with the School of Computer Science and Engineering, Wuhan Institute of Technology, Wuhan 430070, China (e-mail: yananli@wit.edu.cn).

Feng Yang is with the School of Biomedical Engineering, Southern Medical University, Guangzhou 510515, China (e-mail: yangf@smu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMM.2020.3037538.

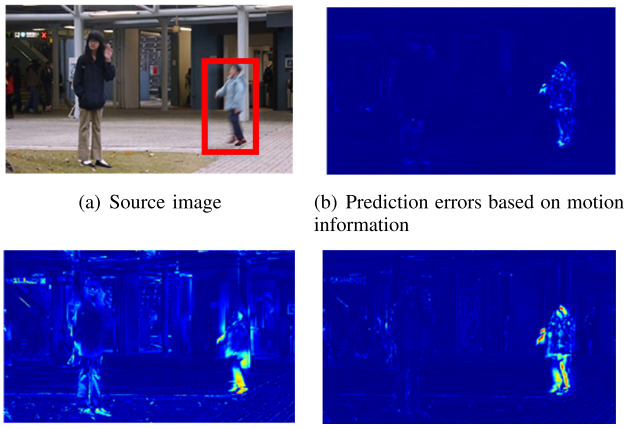
Digital Object Identifier 10.1109/TMM.2020.3037538

I. Introduction

THE dramatic increase of uncontrolled surveillance videos leads to the problem of wasting huge time to observe the meaningless video contents [1]. Effective and real-time anomaly detection approach can essentially help to alleviate this problem [2]–[4]. However, accurate anomaly detection is still remaining as a challenging visual task mainly due to that the boundary between normal and anomaly events is somewhat vague. Generally, an abnormal event can be defined as the instance that does not conform to expected normal patterns [5]. Thus, this task can be accordingly regarded as a self-supervised problem of identifying unexpected events given the normal training samples [2], [6].

Till now, a large number of efforts with self-supervised learning have been paid to detect anomalies by fitting the reconstruction model or prediction model upon the training data [2], [7]. High reconstruction errors or prediction errors tend to indicate anomalies in test samples [8]–[10]. Early anomaly detection methods [11], [12] are mainly built on hand-crafted spatial-temporal visual features extracted from image cells or video volumes. The insufficient descriptive power of the handcraft feature limits the performance. Most recently, with the application of deep learning technology [12]–[15], the performance of anomaly detection has been remarkably leveraged. The main-stream deep learning manners [2], [10] choose to rely on an autoencoder network. They generally resort to using only a single encoder to seek the variations within the videos. However, it is hard for a single encoder to simultaneously deal with all patterns (e.g., motion and content patterns) in videos. The main reasons are that: (1) the challenges of anomaly detection are not only the uncertainty of anomaly patterns, but also different patterns essentially lead to different impacts [16], [17]; (2) due to the lack of accurate labels of normal and abnormal samples, how to detect anomalies is decided by the subjective understanding of models. Thus, as a self-supervised task, different models will prefer to mine different patterns according to their characteristics. We call this problem as a pattern bias problem.

Aiming to illustrate the problem, we provide different reconstruction/prediction error maps in Fig. 1. Shown in Fig. 1(b), based on motion patterns of previous consecutive frames, the prediction errors provided by Liu *et al.* [2] pay attention to the abnormal motion pixels around the running boy. Because of color uniformity, which will weaken the motion impacts, the prediction errors of boy's clothes are small. Oppositely, based on content patterns, the reconstruction errors, which are



(a) Source image (b) Prediction errors based on motion information

(c) Reconstruction errors based on content information (d) Reconstruction errors generated by MESDnet

Fig. 1. Comparison of the prediction/reconstruction errors at frame 512 on the Avenue dataset. The abnormal event is a running boy marked by the red box.

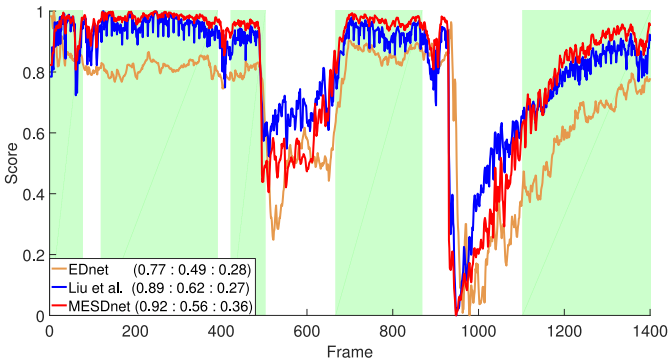


Fig. 2. Learned scores of a video sequence on the CUHK-Avenue dataset [11] by the fully convolutional network [18] (i.e., EDnet), Liu *et al.* [2] and the proposed MESDnet. Peak Signal to Noise Ratio (PSNR) is applied to provide the evaluation score. A low score means anomaly and vice versa. Green regions represent the normal frames. The legend includes the average of normal scores, the average of abnormal scores and the gap between normal and anomaly. It shows that MESDnet achieves the largest gap, which benefits anomaly detection [2].

produced by a traditional auto-encoder [18] denoted as EDnet using the current frame as the input, generally focus on abnormal appearances. However, a large variety of appearances will lead to reconstruction errors at normal pixels. The reconstruction error map of EDnet is presented in Fig. 1(c). Shown in Fig. 1(d), MESDnet integrates the strengths of different models to achieve high errors at abnormal pixels and low errors at normal ones. Furthermore, based on Peak Signal to Noise Ratio (PSNR), learned score curves of the above models are given in Fig. 2. It can be observed that: (1) EDnet is sensitive to abnormal events, but its results at normal frames are unstable; (2) Liu *et al.* [2] achieve high and smooth scores at normal frames, but its scores of abnormal frames are weak. To overcome the aforementioned limitations, Xu *et al.* [16] propose auto-encoder networks to learn feature representations of both content and motion patterns, respectively. However, the auto-encoder networks are only used to extract the appearance representations instead of hand-crafted features, and it is not end-to-end trainable.

In order to alleviate the pattern bias problem, we propose a Multi-Encoder Single-Decoder network, denoted as MESDnet, for anomaly detection in videos through independently characterizing motion and content patterns. By calculating the differences between adjacent frames [19], the motion encoder provides motion features for the decoder. Moreover, the motion residual connections are built between the motion encoder and the decoder. It can prevent information loss after the pooling operations. The content encoder consists of two pathways. The first content encoder is designed to extract the content features of the current frame I_t. To avoid learning an identity function in the task of homologous image reconstruction [20] (i.e., reconstructing $\hat{I}_t$ from I_t), content residual connections are replaced by the connections from the second content encoder to the decoder. Inspired from [21], the last observed frame I_{t-1} also can provide the content features residual connections because neighboring frames have similar content features. Thus, the second content encoder is designed to extract the content features from the last observed frame I_{t-1} and provide the content residual connections to the decoder. These two content pathways share parameters. The frame reconstruction is then achieved by fusing the motion and content information using the decoder.

In the experimental section, a simple yet efficient estimation method is introduced. Generally, multiple frames are used to extract the motion pattern [2], [6]. Intuitively, if the current frame contained abnormal events, it would impact the reconstruction quality of the current frame and its following consecutive frames. It infers that both the current frame and its following consecutive frames can provide the cues to detect anomalies at the current frame. Therefore, we propose a linear interpolation method to fuse the reconstruction errors of the current frame and its following consecutive frames.

The main contributions of this paper are as follows:

- For the task of anomaly detection with self-supervised learning, we propose a Multi-Encoder Single-Decoder network, denoted as MESDnet, which takes advantage of different encoders to mine different patterns, respectively. It can alleviate the pattern bias problem caused by the subjective understanding of different models.
- MESDnet is an end-to-end trainable network that can learn to decompose motion and content without separate training for anomaly detection and reduces the task of anomaly detection to effectively reconstructing the current frame by the observed content and motion.
- A new criterion of anomaly detection based on the consecutive frames is recommended in the experimental section.

The remainder of this paper is organized as follows: The related work is introduced in Sec. II. Then the architecture overview, the network details, and the objective function are illustrated in Sec. III, IV and V, respectively. The testing procedure is described in Sec. VI. The experiments and discussions are conducted in Sec. VII. Sec. VIII concludes the whole paper.

II. Related Work

Great achievements in anomaly detection [22]–[30] have been made in the past years. In this section, we will review hand-crafted feature based anomaly detection and deep learning based anomaly detection, respectively.

A. Anomaly Detection Based on Hand-Crafted Features

For anomaly detection, early hand-crafted feature based works usually utilize low-level trajectory features to represent the normal motion patterns [31], [32]. However, it is difficult for these methods to achieve robust performance in complex or crowded scenes with multiple occlusions and shadows. Instead of trajectory features, hand-crafted spatial-temporal features, such as the histogram of oriented flows (HOF) [33], the histogram of oriented gradients (HOG) [34] are widely used in anomaly detection as the representation of motion and content. Zhang *et al.* [35] formulate anomaly detection as a positive and unlabeled learning problem to learn a detector of anomaly events. Based on spatial-temporal features, Markov random field (MRF) [36], the mixture of probabilistic PCA (MPPCA) [37] and Gaussian mixture model [38] are employed to model the motion and content patterns. In view of the assumption that normal patterns can be described as a linear combination of a learned dictionary, sparse coding and dictionary learning are also widely researched to encode the regular patterns [9]. However, the hand-crafted representations of both motion and content are not good enough for handling all patterns in videos.

B. Anomaly Detection Based on Deep Learning

Recently, deep learning methods have won a great success in many computer vision tasks [13], [14], [39]–[41] as well as anomaly detection [5], [8], [10], [42]. In this section, we will focus on anomaly detection with self-supervised learning. Generally, 3D convolutional neural networks are effective in modeling motion and content simultaneously [43]. A 3D convolutional Auto-Encoder (Conv-AE) is designed to model the normal patterns in regular frames [10]. Combining advantages of convolutional neural networks (CNN) and long short term memory (LSTM) [44], Luo *et al.* [45] propose a convolutional LSTM Auto-Encoder (ConvLSTM-AE) to model normal appearance and motion patterns simultaneously. Considering the temporally coherent anomaly probability, A sparse coding based method within a stacked RNN framework is introduced to model the normal patterns. Different from reconstructing the frame, Lie *et al.* [2] utilize a prediction model to predict the next frame and the prediction error is defined to estimate the anomaly probability. Due to the complexities of anomaly detection, it is burdensome for a single encoder to handle all the different patterns in videos simultaneously. Multiple Auto-Encoder networks for learning feature representations of both content and motion patterns are designed in [16] to overcome the aforementioned limitation. Nevertheless, the encoders are only used to extract the deep learning features instead of hand-crafted features, and it is not an end-to-end trainable network. In contrast with [16], we aim to fuse the motion and content information using multi-encoder architecture in our MESDnet, which can be trained in an end-to-end fashion.

III. Algorithm Overview

In this section, we formally define the role of all components in MESDnet. Let I_t represent the t-th frame in a video. For the task of anomaly detection, the objective is to reconstruct the t-th frame $\hat{I}_t$ given the input frames from $I_{t-\triangle t}$ to I_t, where $\triangle t$ represents the number of previous images. The architecture of the proposed method is described in Fig. 3, and the components in MESDnet are listed as follows:

Motion Encoder and Motion Residual: The motion encoder takes differential image inputs between the neighboring frames starting from $I_{t-\triangle t}$ and ending at I_{t-1}. The hidden representation m_t is provided to model the observed motion pattern. Inspired from [2], the observed motion in the previous frames can achieve high discrimination performance, which can avoid obtaining similar reconstruction errors of normal and abnormal events. Thus, the differential image between I_{t-1} and I_t is not included in the input of the motion encoder. The motion residual takes the features at every scale right before pooling to compute residuals $mRes_t$.

Content Encoder and Content Residual: Intuitively, the content information extracted from frame I_t is the best choice for reconstructing frame $\hat{I}_t$. The content encoder with the input of the current frame I_t is a homologous-image-based encoder (i.e., reconstructing $\hat{I}_t$ from I_t). If there was a direct connection between $\hat{I}_t$ and I_t, the other part of the model would be ignored. Moreover, Lukas *et al.* [20] introduce that an auto-encoder would just learn the identity function if there was no compression. Thus, to avoid learning an identity function, we do not design the residual connection between the content encoder of the current frame and the decoder. The reason is that, if we get an identity model for all frames, it would be useless for anomaly detection. However, without the residual connection, the reconstruction quality of low-level details (e.g., edge) will be unsatisfactory due to the information loss in the procedure of feature compression(i.e., pooling) [46]. This problem will lead to high reconstruction errors at normal frames, which make against anomaly detection. Therefore, the residual connection for reserving the details is essential but can not be added between the content encoder of the current frame and the decoder. In order to alleviate the problems, the content encoder of the last observed frame I_{t-1} is introduced to provide similar detail information to the decoder, because adjacent frames have similar information in videos [21]. The content residual connection is linked between the content encoder of the last observed frame and the decoder.

According to the above analyses, the content encoder takes the current frame I_t and the last observed frame I_{t-1} as inputs and consists of two pathways which share the same encoding parameters. One pathway is the content encoder of the current frame I_t and outputs the hidden representation c_t. The other one is the content encoder of the last observed frame I_{t-1} and output c_{t-1}. Moreover, the content residual is introduced from the encoding features of I_{t-1} at every scale before pooling and computes residuals $cRes_{t-1}$.

Combination Layers and Decoder: The outputs from both encoding pathways and residual connections, m_t, c_t, c_{t-1}, $mRes_t$ and $cRes_{t-1}$, are fused into a decoder to produce a pixel-level frame reconstruction $\hat{I}_t$.

The details of each component in our MESDnet is described in the following section.

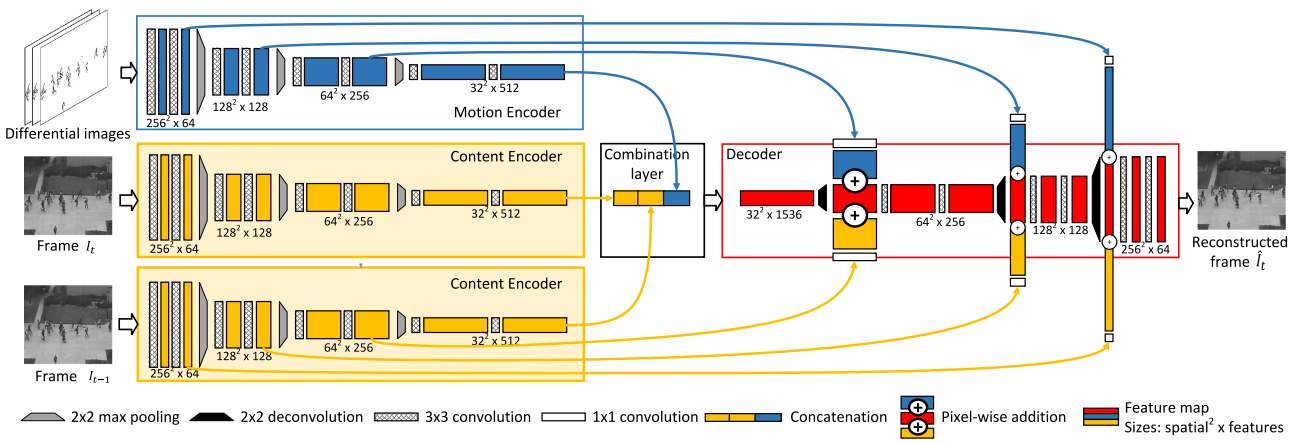


Fig. 3. The architecture of our multi-encoder single-decoder network (MESDnet). Through the motion encoder and the content encoder, our networks extract the motion information from several differential images and the content information from the observed frames. The outputs of the encoders are combined as the input of the decoder, which is designed to reconstruct frame $\hat{I}_t$. To prevent information loss after the pooling operation, we adopt the residual connections to transfer the information from the encoders to the decoder.

IV. Multi-Encoder Single-Decoder Network

This section describes the detailed configuration of the proposed multi-encoder single-decoder network (MESDnet), including the motion encoder and its residual, the content encoder and its residual, the combination layers and the decoder.

A. Motion Encoder and Motion Residual

The motion encoder models the temporal patterns from $\triangle t$-1 consecutive differential images and outputs motion features by

$$m_t = g^m(\triangle I_{t-1}, \ldots, \triangle I_{t-\triangle t+1}), \tag{1}$$

where $\triangle I_{t'} = I_{t'} - I_{t'-1}, t' \in \{t-\triangle t+1, t-\triangle t+2, \ldots t-1\}$ represents the element-wise subtraction between neighboring frames, and m_t is the feature tensor encoding the motion patterns across the image difference inputs. $g^m(.)$ is a fully-convolutional operation as the encoder of U-net [47] which can model local motion patterns [21].

To prevent the information loss after the pooling operations in the encoder, we adopt the residual connections [48] to transfer motion features at every scale into the decoder layers after deconvolution operations. The residual feature at layer l is calculated as

$$mRes_t^l = g^r(m_t^l)^l, \tag{2}$$

where $mRes_t^l$ and m_t^l represent the residual output and the motion feature at layer l respectively, and $g^r(.)^l$ is the residual function at layer l implemented as a 1×1 convolution operation.

B. Content Encoder and Content Residual

The spatial features are extracted by the content encoder from the current frame I_t and the previous frame I_{t-1}. It contains two separated encoders sharing the same parameters, and produces content features by

$$c_t = g^c(I_t), \tag{3}$$

$$c_{t-1} = g^c(I_{t-1}), \tag{4}$$

where c_t and c_{t-1} are the features, which encode the spatial content patterns of I_t and I_{t-1}, respectively; $g^c(.)$ is implemented as the encoder of U-net [47] that extracting content features from single frame.

Aiming to avoid learning an identity function in the task of homologous image reconstruction for anomaly detection [20], the pathway of the current frame only provides the high-level hidden features c_t. To prevent the information loss in the content encoder, residual connections from the encoder of the last observed frame I_{t-1} is utilized to communicate the content features at every scale into the decode layers after deconvolution operations. The motivation is that the neighboring frames have similar content information. The residual feature at layer l is computed as

$$cRes_{t-1}^l = g^r(c_{t-1}^l)^l, \tag{5}$$

where $cRes_{t-1}^l$ and c_{t-1}^l represent the residual output and the content feature at the l-th layer, respectively; $g^r(.)^l$ is the residual function.

In our model, an asymmetric architecture is employed for the motion and content encoders. As the most important clue, I_t and I_{t-1} are taken by the content encoders to reconstruct spatial layout. But the content encoder has no information about dynamics. On the other side, several differential images are the input of the motion encoder, which focuses on modeling the temporal patterns occurring over time. This asymmetric architecture can naturally fuse the content and motion in an end-to-end network by learning motion and content decomposition.

C. Combination Layers and Decoder

The output features, m_t, c_t and c_{t-1}, are high-level representations of motion and content, respectively. Given these features, the frame $\hat{I}_t$ can be reconstructed by the decoder. Here, the high-level representations are combined by

$$f_t = [m_t, c_t, c_{t-1}], \tag{6}$$

where [.] represents the concatenation operation in the channel dimension, and f_t is the combination feature. Then the decoder in MESDnet reconstruct $\hat{I}_t$ from f_t by

$$\hat{I}_t = g^d(f_t, mRes_t, cRes_{t-1}), \tag{7}$$

where $mRes_t$ and $cRes_{t-1}$ denote the residual connections which link the motion and content encoders before pooling and the decoder after deconvolution, respectively. The deconvolution network [2] is utilized for our decoder network $g^d(.)$, which consists of multiple successive operations of deconvolution and convolution. After each deconvolution operation, a pixel-wise addition operation will be implemented between the motion residual feature, the content residual feature and the deconvolution feature. Finally, tanh(.) is utilized as the activation layer after the last convolution layer.

V. Objective Function

In the objective function, we adopt the constrains on intensity, gradient, motion and adversarial training as

$$\begin{aligned}\mathcal{L}_{\mathcal{G}} = {} & \lambda_{int} L_{int}(\hat{I}_t, I_t) + \lambda_{gd} L_{gd}(\hat{I}_t, I_t) \\ & + \lambda_{op} L_{op}(\hat{I}_t, I_t, I_{t-1}) + \lambda_{adv} L^{\mathcal{G}}_{adv}(\hat{I}_t),\end{aligned} \tag{8}$$

where $L_{int}(\hat{I}_t, I_t)$, $L_{gd}(\hat{I}_t, I_t)$, $L_{op}(\hat{I}_t, I_t)$ and $L^{\mathcal{G}}_{adv}(\hat{I}_t)$ represent the loss of intensity, gradient, motion and generator, respectively. λ_{int}, λ_{gd}, λ_{op} and λ_{adv} are the weight parameters, separately.

When we train the discriminative network $\mathcal{D}$, the following loss function is used:

$$L_{\mathcal{D}} = L^{\mathcal{D}}_{adv}(\hat{I}_t, I_t), \tag{9}$$

where $L^{\mathcal{D}}_{adv}(\hat{I}_t, I_t)$ is the discriminator loss. Aiming to train the network, we normalize the intensity of pixels to $[-1, 1]$ and resize the input size of each frame to 256×256. $\triangle t$ and the mini-batch size are both set to 4. λ_{int}, λ_{gd}, λ_{op} and λ_{adv} slightly vary from datasets and we can also easily set them as 1.0, 1.0, 2.0 and 0.05, respectively [2]. Next, we will introduce the constrains in detail.

Constrains on intensity and gradient: Following the works [49], the intensity and gradient loss are used in the image space to reconstruct $\hat{I}_t$. The intensity loss constrains the pixel-level similarity between the reconstructed image $\hat{I}_t$ and its ground truth I_t. The gradient loss can make the reconstructed image sharp. The intensity loss $L_{int}(\hat{I}_t, I_t)$ is defined as follow:

$$L_{int}(\hat{I}_t, I_t) = ||\hat{I}_t - I_t||_2^2, \tag{10}$$

where the ℓ_2 distance is used for the intensity loss. Moreover, the gradient loss $L_{gd}(\hat{I}_t, I_t)$ is defined as follows:

$$\begin{aligned}& L_{gd}(\hat{I}_t, I_t) \\ & = \sum_{i,j} (\left\| \left| \hat{I}_t(i,j) - \hat{I}_t(i-1,j) \right| - \left| I_t(i,j) - I_t(i-1,j) \right| \right\|_1 \\ & + \left\| \left| \hat{I}_t(i,j) - \hat{I}_t(i,j-1) \right| - \left| I_t(i,j) - I_t(i,j-1) \right| \right\|_1),\end{aligned} \tag{11}$$

where i and j denote the spatial index of the t-th frame; $|*|$ represent the absolute value of $*$; and $||*||_1$ is the ℓ_1 distance.

Constrain on motion: Because the optical flow is sensitive to a slight change in terms of the intensity of all pixels [50], optical flow loss is applied as [2]. Based on Flownet [51], the loss of optical flow is defined as follows:

$$L_{op}(\hat{I}_t, I_t, I_{t-1}) = ||Flow(\hat{I}_t, I_{t-1}) - Flow(I_t, I_{t-1})||_1, \tag{12}$$

where $Flow(A, B)$ represents the optical flow between A and B, and it is pre-trained on a synthesized dataset [51].

Constrain on adversarial training: Generative adversarial network (GAN) has been widely used in computer vision tasks [52]. Generally, GAN is composed of a discriminative network $\mathcal{D}$ and a generative network $\mathcal{G}$. In our proposed MESDnet, we treat the reconstruction network as the generator $\mathcal{G}$, which learns to generate frames that are hard to be classified by the discriminative network $\mathcal{D}$. Following [2], we use a patch discriminator for $\mathcal{D}$. It means that each input frame will be divided into several patches and each patch will have a corresponding output score. The loss in term of adversarial training [2] is introduced as follows:

$$\begin{aligned}L^{\mathcal{D}}_{adv}(\hat{I}_t, I_t) = {} & \sum_{i,j} \frac{1}{2} MSE(\mathcal{D}(I_t(i,j)), 1) \\ & + \sum_{i,j} \frac{1}{2} MSE(\mathcal{D}(\hat{I}_t(i,j)), 0),\end{aligned} \tag{13}$$

$$L^{\mathcal{G}}_{adv}(\hat{I}_t) = \sum_{i,j} \frac{1}{2} MSE(D(\hat{I}_t(i,j)), 1), \tag{14}$$

where i and j denote the spatial patch indexes of the t-th frame, and $MSE(*)$ is a Mean Square Error function defined as follows:

$$MSE(\hat{A}, A) = (\hat{A} - A)^2, \tag{15}$$

where A takes values in $\{0, 1\}$ and $\hat{A} \in [0, 1]$.

VI. Traditional Anomaly Detection on Testing Data

Here, we will introduce the details of the traditional anomaly detection on the testing data and adopt this method in all experiments except the last one for a fair comparison. In the last experiment, an alternative anomaly detection method based on the linear interpolation operation will be introduced and evaluated.

In the testing phase, $\triangle t$-1 differential images, frame I_{t-1} and frame I_t are fed into MESDnet. With one forward pass, we can obtain the reconstructed result $\hat{I}_t$. Generally, Peak Signal to Noise Ratio (PSNR) [49] can be used to estimate the reconstruction quality of each frame. PSNR is defined as follows:

$$P(t) = 10 \log_{10} h_t, \tag{16}$$

where

$$h_t = \frac{[\max_{\hat{I}_t}]^2}{\frac{1}{N} \sum_{i,j} (I_t(i,j) - \hat{I}_t(i,j))^2}. \tag{17}$$

where i and j represent the spatial index of $\hat{I}_t$ and I_t, $[\max_{\hat{I}_t}]$ is the maximum value of $\hat{I}_t$, and N is the number of pixels. Low $P(t)$ means that it is more likely to be abnormal and vice versa. After getting all frames' PSNR scores in each testing video, following the works [2], [49], scores of all frames are normalized to [0, 1] as follows:

$$S(t) = \frac{P(t) - \min_{t' \in [1,T]} P(t')}{\max_{t' \in [1,T]} P(t') - \min_{t' \in [1,T]} P(t')}. \quad (18)$$

where T is the number of frames in a video, and $\min_{t' \in [1,T]} P(t')$ and $\max_{t' \in [1,T]} P(t')$ represent the minimum and the maximum PSNR values in the video, respectively. Therefore, given a threshold, we can distinguish whether a frame is normal or abnormal according to its score $S(t)$.

VII. Experiments

In the experiments, we evaluate the anomaly detection performance[1] and the time consumption of the proposed MESDnet to demonstrate its effectiveness and efficiency. Three publicly anomaly detection datasets (i.e., the CUHK Avenue dataset [11], the UCSD Pedestrian dataset [38] and the ShanghaiTech Campus dataset [6]) are employed for training and testing. On the CUHK Avenue dataset [11], 16 videos in the training set are used to train the models, and 21 testing videos are picked out for the performance evaluation. There are 47 abnormal events in this dataset, including throwing objects, loitering and running. The size of people may change significantly. The UCSD dataset contains the UCSD Pedestrian 1 dataset and the UCSD Pedestrian 2 dataset. They are denoted as Ped1 and Ped2, respectively. 34 training videos and 36 testing videos with 40 abnormal events are set in Ped1, while Ped2 has 16 training videos and 12 testing videos with 12 irregular events. The abnormal events are mainly triggered by bicycles and cars. Following [54], we exclude Ped1 from the estimation, because the frame resolution of 158×238 is significantly low. The other dataset is the ShanghaiTech Campus dataset [6], [53], which includes 330 training videos and 107 test ones. Different from the aforementioned datasets, the ShanghaiTech dataset consists of 13 different scenes, which make it more challenging.

The experimental section is organized as follows: the evaluation metric and the implementation details are introduced in Sec. VII-A and VII-B, respectively. The comparison experiments with existing state-of-the-art methods are given in Sec. VII-C. In Sec. VII-D, we will investigate the performances of different encoder combinations. Then, the qualitative evaluation of anomalous event detection will be illustrated in Sec. VII-E, and the parameter sensitivity investigation is shown in Sec. VII-F. Sec. VII-G provides the analyses of the time consumption. Finally, a new evaluation method of anomaly detection is introduced and evaluated in Sec. VII-H.

[1]Following [53], this paper pays attention to the frame-level anomaly detection

TABLE I
AUC Comparison Between MESDnet and the Stat-of-the-Art Methods on the Avenue, Ped2 and ShanghaiTech Datasets. The Best Result Is Marked in Bold

	Avenue	Ped2	ShanghaiTeck
MPPCA [37]	N/A	69.3%	N/A
MPPCA+SFA [38]	N/A	61.3%	N/A
MDT [38]	N/A	82.9%	N/A
Conv-AE [10]	80.0%	85.0%	60.9%
ConvLSTM-AE [45]	77.0%	88.1%	N/A
AMDN [16]	N/A	90.8%	N/A
DeepAppearance [55]	84.6%	N/A	N/A
Unmasking [26]	80.6%	82.2%	N/A
TSC [6]	80.6%	91.0%	68.0%
Stacked RNN [6]	81.7%	92.2%	68.0%
Liu et al. [2]	85.1%	95.4%	72.8%
MemAE [56]	83.3%	94.1%	71.2%
MESDnet	**86.3**%	**95.6**%	**73.2**%

A. Evaluation Metric

Based on Eqn. 18, we can estimate the score of frame t and identify whether an abnormal event occurs. Given a fixed threshold, the frame can be categorized as an anomaly frame if its score is lower than the threshold. Obviously, a high threshold will lead to a high false-negative ratio and a low one may produce more false alarms. In the literature of anomaly detection [2], [11], [38], the Receive Operation Characteristic (ROC) by gradually changing the threshold is introduced to get the performance curves of different anomaly detection methods. And the Area Under Curve (AUC) is computed as a performance evaluation. A higher value means better performance and vice versa. In this paper, following the work [2], we also adopt AUC as the performance evaluation.

B. Implementation Details

In the experiments, we use $\triangle t = 4$ consecutive frames to compute the differential images as the input of the motion encoding path. All input frames are resized to 256×256, and the intensity of pixels are normalized to $[-1, 1]$. In MESDnet, the number of layers is set to 4. All the experiments are conducted on the same PC with Xeon(R) W-2145 CPU and TITAN Xp GPU.

C. Comparison With Existing Methods

Here, we compare MESDnet with different state-of-the-art methods: MPPCA [37], MPPCA+SFA [38], MDT [38], Conv-AE [10], ConvLSTM-AE [45], AMDN [16], DeepAppearance [55], Unmasking [26], TSC [6], Stacked RNN [6], Liu *et al.* [2] and MemAE [56]. We list the AUC performance of different methods on these datasets in Table I. It can be seen that the performance of MESDnet is generally better than other methods.

D. Comparison of Different Encoder Combinations

In this section, different encoder combinations are conducted to evaluate the effects of different encoders on the Avenue and Ped2 datasets. This experiment is not conducted on the ShanghaiTeck dataset because of its significantly large size. For each combination, the constrains on intensity, gradient, motion and

TABLE II
AUC COMPARISON OF MESDNET WITH DIFFERENT ENCODER COMBINATIONS

	Avenue	Ped2
EDnet	82.8%	84.8%
MESDnet-simple	83.9%	93.0%
MESDnet-prediction	84.8%	95.4%
MESDnet	**86.3%**	**95.6%**

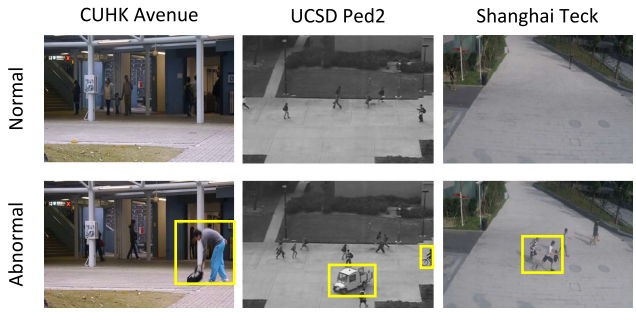


Fig. 4. Samples in the CUHK Avenue, UCSD, ShanghaiTeck datasets. The samples in the first row are normal samples and others denote anomalies in abnormal frames. The abnormal events are marked in yellow bounding boxes.

adversarial training are used to supervise the training procedure. The combinations include: (1) the content encoder of the current frame; (2) the motion encoder and the content encoder of the current frame, which is a simple version of MESDnet; (3) the motion encoder and the content encoder for the last observed frame, which can be treated as a prediction version of MESDnet; (4) all encoders. For conciseness, they are denoted as EDnet, MESDnet-simple, MESDnet-prediction and MESDnet, respectively. We list the AUC performance of different combinations in Table. II. It can be observed that:

• MESDnet-simple, which includes the motion encoder and the content encoder for the current frame, get a higher AUC score than EDnet. It means that the motion pattern is an essential cue for anomaly event detection.

• MESDnet-prediction, which contains the motion encoder and the content encoder of the last observed frame, can be treated as a prediction network. It also gains a high AUC score, because the content of the last observed frame is similar to that of the current frame. Moreover, the motion residual and the content residual can prevent information loss after the pooling operations in the encoders.

• MESDnet, which fuses the observed motion and content, achieves the best performance. It outperforms EDnet by about 4% in terms of AUC on the Avenue dataset and about 11% on the Ped2 dataset. The reason is that multiple encoders can model different regular patterns (e.g., motion and content), respectively. Moreover, the performance of MESDnet is higher than that of MESDnet-prediction. It means that it is hard for the prediction model to handle complex patterns only depending on the previous information. Especially on the Avenue dataset, the content of the current frame helps the network achieve 1.5% improvement.

E. Qualitative Evaluation of Anomalous Event Detection

Fig. 5 shows the scores as a function of frame number. In this figure, the low score means anomaly and vise versa. As aforementioned, EDnet, MESDnet-simple, MESDnet-prediction and MESDnet represent the encoder-decoder network for the current frame, the combination between the motion encoder and the content encoder of the current frame, the combination between the motion encoder and the content encoder of the last observed frame, and the combination of all encoders, respectively. It can be observed that:

• MESDnet gets the largest gap between normal and anomaly while obtaining the highest average of normal scores. The large gap benefits detecting anomalies and the high average of normal scores can reduce false alarm of abnormal events. Thus, MESDnet can achieve the best performance shown in Table II.

• Both MESDnet and MESDnet-prediction have high averages of normal scores. The main reason is that the residual connection between the content encoder and the decoder helps the model to achieve high reconstruction quality at normal frames. Oppositely, EDnet and MESDnet-simple get lower averages of normal scores.

• Due to the existence of the content encoder of the current frame, EDnet, MESDnet-simple and MESDnet can produce low abnormal scores. It means that the content information of the current frame is important for detecting anomalies.

F. Parameter Sensitivity Investigation

In MESDnet, the number $\triangle t$ of consecutive images in the motion encoding path and the number L of layers are tunable parameters. The former parameter will impact the effectiveness of encoding the motion information, and the latter is a key share parameter of the motion encoder, the content encoder and the decoder in the model. Here, we set the parameters $\triangle t \in \{2, 3, 4, 5\}$ and $L \in \{2, 3, 4, 5\}$. The detailed results on the Avenue and Ped2 datasets are listed in Table III and IV, respectively. It can be observed that:

• In MESDnet, the consecutive images are used as the input of differential images. More differential images can introduce more normal motion patterns into the encoding path. The MESDnet with $\triangle t = 4$ achieves the best performance of anomaly detection, because fewer frames could not effectively handle the motion pattern, and more frames might introduce unexpected noises.

• MESDnet with $L = 4$ gets the best results. Generally, deep layers obtain the high-level semantic features while shallow layers focus on the low-level spatial features. The performance of MESDnet with $L = 2$ or $L = 3$ is not good, because it is not enough for low-level spatial features to encode normal patterns. When the model is too deep, such as $L = 5$, its performance is also not satisfactory. The reason may be that serious information losses would lead to construction errors.

G. Time Consumption

In this section, we will analyze the time and space complexity of MESDnet. For each convolutional layer and deconvolution layer, the time complexity can be calculated as

$$T \sim O(H_f W_f \cdot H_k W_k \cdot C_k C_f), \tag{19}$$

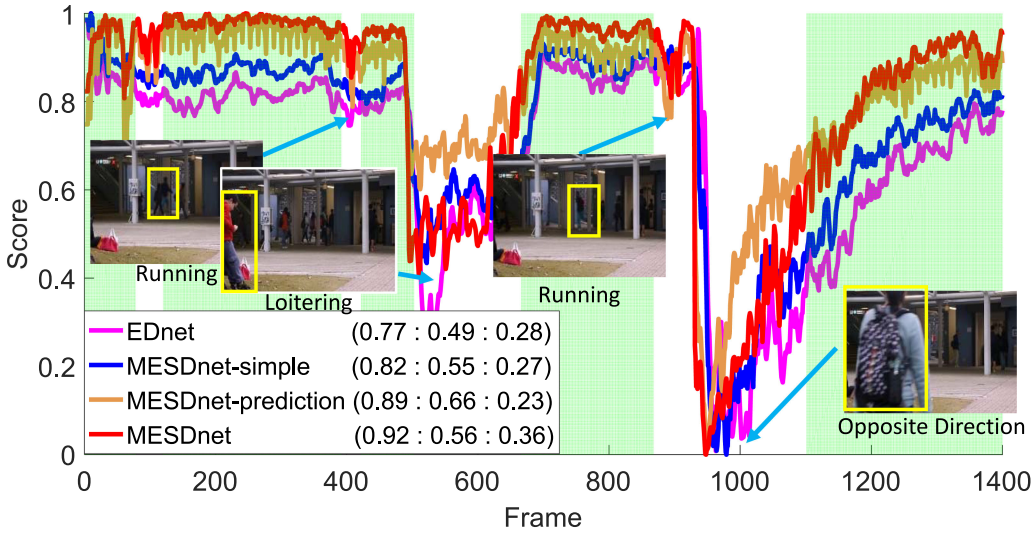


Fig. 5. Scores of a sequence on the Avenue dataset. A low score means anomaly and vice versa. Green regions represent the normal frames, and the anomaly events are marked in yellow bounding boxes. The legend includes the average of normal scores, the average of abnormal scores and the gap between normal and anomaly. Best viewed in color.

TABLE III
AUC COMPARISON OF MESDNET WITH DIFFERENT NUMBERS $\triangle t$ OF CONSECUTIVE IMAGES

$\triangle t$	Avenue	Ped2
2	61.2%	95.3%
3	84.9%	95.2%
4	**86.3%**	**95.6%**
5	85.5%	94.1%

TABLE IV
AUC COMPARISON OF MESDNET WITH DIFFERENT NUMBERS L OF LAYERS

L	Avenue	Ped2
2	83.5%	91.2%
3	84.1%	**95.6%**
4	**86.3%**	**95.6%**
5	84.7%	95.3%

TABLE V
TIME AND SPACE COMPLEXITY OF MESDNET. M, G, AND FLOPS REPRESENT MILLION,GILLION AND THE NUMBER OF FLOATING-POINT OPERATIONS PER SECOND, RESPECTIVELY

Time complexity (FLOPS)	71 G
Space complexity	80 M

TABLE VI
INPUT-OUTPUT RELATIONSHIP OF MESDNET WITH $\triangle t = 4$

Input	Output
$I_{t-4}, I_{t-3}, I_{t-2}, I_{t-1}, I_t$	$\hat{I}_t$
$I_{t-3}, I_{t-2}, I_{t-1}, I_t, I_{t+1}$	$\hat{I}_{t+1}$
$I_{t-2}, I_{t-1}, I_t, I_{t+1}, I_{t+2}$	$\hat{I}_{t+2}$
$I_{t-1}, I_t, I_{t+1}, I_{t+2}, I_{t+3}$	$\hat{I}_{t+3}$
$I_t, I_{t+1}, I_{t+2}, I_{t+3}, I_{t+4}$	$\hat{I}_{t+4}$

where H_f and W_f are the height and width of feature maps, respectively; and H_k and W_k are height and width of the kernel, respectively. C_k and C_f are the number of channels for the kernel and output feature maps, separately. Therefore, for the whole model, its time complexity is defined as

$$T \sim O\left(\sum_{i=1}^{n} H_f^i W_f^i \cdot H_k^i W_k^i \cdot C_k^i C_f^i\right), \tag{20}$$

Similarly, the following formula is used to evaluate the space complexity, which consists of parameters and output feature maps.

$$S \sim O\left(\sum_{i=1}^{n} H_k^i W_k^i \cdot C_k^i C_f^i + H_f^i W_f^i \cdot C_f^i\right), \tag{21}$$

According to Eqn. 20 and 21, the time and space complexity of the proposed MESDnet are listed in Table V.

The processing power of TITAN Xp GPUs is 12100 GFLOPS. Thus, the proposed method can theoretically achieve real-time performance. MESDnet is implemented with Tensorflow [57] and tested on the aforementioned datasets. All experiments are estimated on Nvidia Geforce TITAN Xp GPUs with Intel(R) Xeon(R) W-2145 3.7GHz CPUs. The average running time is about 30fps, which includes both frame reconstruction and anomaly detection.

H. A New Method of Anomaly Detection

As many existing methods [2], [6], [10], MESDnet adopts multiple neighboring frames to extract the motion pattern. Thus, anomaly events would not only lead to the reconstruction error of the current frame, but also lead to the reconstruction error of its following consecutive frames. The input-output relationship of MESDnet with $\triangle t = 4$ is given in Table VI. I and $\hat{I}$ represent the original frame and the reconstructed frame, respectively. Their subscripts (e.g., t) are the frame index. The input frames will be pre-processed for the inputs of the motion encoder and the content encoder. Shown as Table VI, frame I_t will impact the

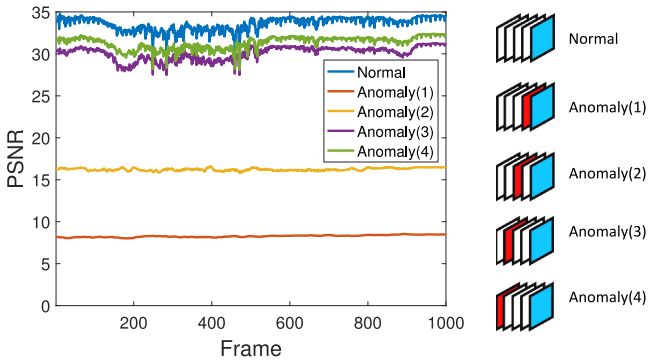


Fig. 6. Sample of the scores of different combinations between the normal frames and the abnormal ones. We choose a sequence from the training set on the CUHK Avenue dataset and manually design an abnormal frame using the average value of the whole dataset. The white, blue and red frames are the normal frames, the reconstructed frames and the abnormal frames, respectively. According to the location of the abnormal frame, they are denoted as Normal, Anormal(1), Anormal(2), Anormal(3) and Anormal(4), separately. The curves in the left column represent the PSNR scores of all reconstructed frames. Low PSNR scores denote the bad reconstruction quality.

reconstruction quality of $\hat{I}_t$, $\hat{I}_{t+1}$, $\hat{I}_{t+2}$, $\hat{I}_{t+3}$, and $\hat{I}_{t+4}$. It infers that $\hat{I}_t$, $\hat{I}_{t+1}$, $\hat{I}_{t+2}$, $\hat{I}_{t+3}$, and $\hat{I}_{t+4}$ can provide the cues to detect anomalies at frame I_t. Furthermore, a visualization experiment is shown in Fig. 6. Given the different location of an abnormal frame in each video slice with five frames of a sequence, the PSNR values of reconstructed frames are shown in the left column of Fig. 6. We can observe that (1) an abnormal frame will impact the reconstruction quality of its following consecutive frames; (2) the closer the distance between the abnormal frame and the reconstructed frame, the greater the effect.

According to this phenomenon, we design a linear-interpolation-based method to calculate the score of each reconstructed frame. The score function is defined as follows:

$$S(t) = 10 \log_{10} \hat{h}_t, \tag{22}$$

where

$$\hat{h}_{t'} = \begin{cases} h_{t'}, t' = t + \triangle t \\ \alpha h_{t'} + (1-\alpha)\hat{h}_{t'+1}, t' = t, t+1, \ldots t + \triangle t - 1 \end{cases} \tag{23}$$

where $h_{t'}, t' \in \{t, t+1, \ldots t+\triangle t\}$ is introduced in Eqn. 17; $\triangle t = 4$ is the number of the consecutive frames; and α is empirically set to 0.5 [15]. Low $S(t)$ means that it is more likely to be abnormal and vice versa.

After getting all frame's $S(t)$ scores in each testing video, the scores of all frames are normalized to [0,1] as follows:

$$\hat{S}(t) = \frac{S(t) - \min_{t' \in [1,T]} S(t')}{\max_{t' \in [1,T]} S(t') - \min_{t' \in [1,T]} S(t')}. \tag{24}$$

where T is the number of frames in a video, and $\min_{t' \in [1,T]} S(t')$ and $\max_{t' \in [1,T]} S(t')$ represent the minimum and the maximum $S(t)$ values in the video, respectively. Therefore, given a threshold, we can distinguish whether a frame is normal or abnormal according to its score $\hat{S}(t)$.

TABLE VII
AUC COMPARISON OF DIFFERENT ANOMALY DETECTION METHODS. THE METHODS WITH EQN. 18 ARE MARKED WITH (EQN. 18), AND THE METHODS WITH THE PROPOSED DETECTION METHOD ARE MARKED WITH (EQN. 24)

	Avenue	Ped2	ShanghaiTeck
Gao et al.(Eqn. 18) [2]	85.1 %	95.4%	72.8%
Gao et al.(Eqn. 24) [2]	**86.5**%	**95.6**%	**73.3**%
MESDnet(Eqn. 18)	86.3%	95.6%	73.2%
MESDnet(Eqn. 24)	**87.1**%	**95.8**%	**73.4**%

Based on the traditional evaluation method (i.e., Eqn. 18), the methods of Gao *et al.* [2] and MESDnet with Eqn. 18 are denoted as Gao *et al.*(Eqn. 18) and MESDnet(Eqn. 18), respectively. Gao *et al.*(Eqn. 24) and MESDnet(Eqn. 24) represent the methods with the new evaluation method (i.e., Eqn. 24), separately. The results are listed in Table VII. It can be observed that:

- The AUC scores of MESDnet(Eqn. 24) is higher than that of the methods with Eqn. 18. The improvements of MESDnet(Eqn. 24) range from 0.2% to 0.8%. It verifies that the following consecutive frames can be used to detect anomalies in the current frame.
- The method introduced by Gao *et al.* [2] also obtain the performance improvement while using the proposed testing method. Especially, the improvement on the Avenue dataset is 1.4%.
- The use of consecutive frames' scores may cause a little delay in providing the final score of frame I_t. For example, four consecutive frames will lead to about 100mS delay in a real-time video. In off-line applications and most on-line applications, a delay of 100mS is acceptable.

VIII. CONCLUSION

Anomaly detection in videos is defined to detect the events that do not conform to expected behavior, and this task can be treated as identifying anomaly events given the normal training samples. In this paper, we propose a multi-encoder single-decoder network to reconstruct each frame in videos. The reconstruction error is adopted to estimate the reconstruction quality. Low reconstruction quality infers the existence of abnormal events. The proposed model employs the motion encoder and the content encoder to decompose motion and content without separate training. Moreover, the content encoder includes two pathways for the current frame and the last observed frame, respectively. Then, reconstructing the current frame evolves to decode the extracted current content features into the reconstructed frame by the compensation of observed previous motion and content features. The experimental results on three publicly available datasets show that our method generally outperforms other state-of-the-art methods, which validates the effectiveness of the proposed model. Additionally, a new method of anomaly detection is evaluated in the experimental section.

REFERENCES

[1] M. Sun, A. Farhadi, and S. Seitz, "Ranking domain-specific highlights by analyzing edited videos," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 787–802.

[2] W. Liu, D. L. W. Luo, and S. Gao, "Future frame prediction for anomaly detection—A new baseline," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6536–6545.

[3] K. Xu, X. Jiang, and T. Sun, "Anomaly detection based on stacked sparse coding with intraframe classification strategy," *IEEE Trans. Multimedia*, vol. 20, no. 5, pp. 1062–1074, May 2018.

[4] W. Chu, H. Xue, C. Yao, and D. Cai, "Sparse coding guided spatiotemporal feature learning for abnormal event detection in large videos," *IEEE Trans. Multimedia*, vol. 21, no. 1, pp. 246–255, Jan. 2019.

[5] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, 2009.

[6] W. Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked rnn framework," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 1, no. 2, 2017, pp. 341–349.

[7] J. R. Medel and A. Savakis, "Anomaly detection in video using predictive convolutional long short-term memory networks," 2016, *arXiv:1612.00390*.

[8] Y. S. Chong and Y. H. Tay, "Abnormal event detection in videos using spatiotemporal autoencoder," in *Proc. Int. Symp. Neural Netw.* Springer, 2017, pp. 189–196.

[9] Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 3449–3456.

[10] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 733–742.

[11] C. Lu, J. Shi, and J. Jia, "Abnormal event detection at 150 FPS in MATLAB," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2720–2727.

[12] A. Del Giorno, J. A. Bagnell, and M. Hebert, "A discriminative framework for anomaly detection in large videos," in *Proc. Eur. Conf. Comput. Vis*. Springer, 2016, pp. 334–349.

[13] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *Proc. Eur. Conf. Comput. Vis*. Springer, 2016, pp. 850–865.

[14] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[15] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3074–3082.

[16] D. Xu, Y. Yan, E. Ricci, and N. Sebe, "Detecting anomalous events in videos by learning deep representations of appearance and motion," *Comput. Vis. Image Understanding*, vol. 156, pp. 117–127, Mar. 2017.

[17] V. Kalogeiton, P. Weinzaepfel, V. Ferrari, and C. Schmid, "Joint learning of object and action detectors," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4163–4172.

[18] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[19] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould, "Dynamic image networks for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3034–3042.

[20] R. Lukas *et al.*, "Deep one-class classification," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4390–4399.

[21] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee, "Decomposing motion and content for natural video sequence prediction," 2017, *arXiv:1706.08033*.

[22] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6479–6488.

[23] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, "MVTec AD–a comprehensive real-world dataset for unsupervised anomaly detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9592–9600.

[24] D. T. Nguyen, Z. Lou, M. Klar, and T. Brox, "Anomaly detection with multiple-hypotheses predictions," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 4800–4809.

[25] J. T. Zhou *et al.*, "Anomalynet: An anomaly detection network for video surveillance," *IEEE Trans. Inform. Forensics Secur.*, vol. 14, no. 10, pp. 2537–2550, Oct. 2019.

[26] R. Tudor Ionescu, S. Smeureanu, B. Alexe, and M. Popescu, "Unmasking the abnormal events in video," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2895–2903.

[27] J.-X. Zhong, N. Li, W. Kong, S. Liu, T. H. Li, and G. Li, "Graph convolutional label noise cleaner: Train a plug-and-play action classifier for anomaly detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1237–1246.

[28] R. T. Ionescu, F. S. Khan, M.-I. Georgescu, and L. Shao, "Object-centric auto-encoders and dummy anomalies for abnormal event detection in video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7842–7851.

[29] R. Morais *et al.*, "Learning regularity in skeleton trajectories for anomaly detection in videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11 996–12 004.

[30] S. Ntalampiras, I. Potamitis, and N. Fakotakis, "Probabilistic novelty detection for acoustic surveillance under real-world conditions," *IEEE Trans. Multimedia*, vol. 13, no. 4, pp. 713–719, Aug. 2011.

[31] Q. Sun, H. Liu, and T. Harada, "Online growing neural gas for anomaly detection in changing surveillance scenes," *Pattern Recognit.*, vol. 64, pp. 187–201, 2017.

[32] S. Wu, B. E. Moore, and M. Shah, "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2054–2060.

[33] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Proc. Eur. Conf. Comput. Vis*. Springer, 2006, pp. 428–441.

[34] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 886–893.

[35] J. Zhang, Z. Wang, J. Meng, Y.-P. Tan, and J. Yuan, "Boosting positive and unlabeled learning for anomaly detection with multi-features," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1332–1344, May 2019.

[36] D. Zhang, D. Gatica-Perez, S. Bengio, and I. McCowan, "Semi-supervised adapted hmms for unusual event detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 611–618.

[37] J. Kim and K. Grauman, "Observe locally, infer globally: A space-time mrf for detecting abnormal activities with incremental updates," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2921–2928.

[38] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1975–1981.

[39] B. Luo, H. Li, F. Meng, Q. Wu, and C. Huang, "Video object segmentation via global consistency aware query strategy," *IEEE Trans. Multimedia*, vol. 19, no. 7, pp. 1482–1493, Jul. 2017.

[40] C. Liu, P. Liu, W. Zhao, and X. Tang, "Robust tracking and redetection: Collaboratively modeling the target and its context," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 889–902, Apr. 2018.

[41] S. Tang, Y. Li, L. Deng, and Y. Zhang, "Object localization based on proposal fusion," *IEEE Trans. Multimedia*, vol. 19, no. 9, pp. 2105–2116, Sep. 2017.

[42] K. Xu, T. Sun, and X. Jiang, "Video anomaly detection and localization based on an adaptive intra-frame classification network," *IEEE Trans. Multimedia*, vol. 22, no. 2, pp. 394–406, Feb. 2020.

[43] L. Wu, Y. Wang, L. Shao, and M. Wang, "3D personvlad: Learning deep global representations for video-based person re-identification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3347–3359, Nov. 2019.

[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[45] W. Luo, W. Liu, and S. Gao, "Remembering history with convolutional LSTM for anomaly detection," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2017, pp. 439–444.

[46] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv*. Springer, 2015, pp. 234–241.

[47] R. Olaf, F. Philipp, and B. Thomas, "Ronneberger o, fischer p, brox t. u-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

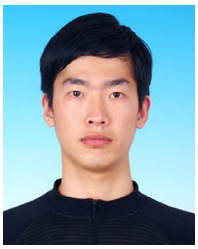
[49] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2813–2821.

[50] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 568–576.

[51] A. Dosovitskiy *et al.*, "Flownet: Learning optical flow with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2758–2766.

[52] E. L. Denton *et al.*, "Deep generative image models using a laplacian pyramid of adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1486–1494.

[53] W. Luo *et al.*, "Video anomaly detection with sparse coding inspired deep neural networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, doi: 10.1109/TPAMI.2019.2944377.
[54] R. Hinami, T. Mei, and S. Satoh, "Joint detection and recounting of abnormal events by learning deep generic knowledge," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3619–3627.
[55] S. Smeureanu, R. T. Ionescu, M. Popescu, and B. Alexe, "Deep appearance features for abnormal behavior detection in video," in *Proc. Int. Conf. Image Anal. Process.*. Springer, 2017, pp. 779–789.
[56] D. Gong *et al.*, "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1705–1714.
[57] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Accessed: Nov. 2019. [Online]. Available: https://www.tensorflow.org



Zhiwen Fang received the B.S. and M.S. degrees from the Automation School of Beihang University, and the Ph.D. degree from the Huazhong University of Science and Technology China. He was the Research Fellow with the Institute of Media Innovation, Nanyang Technological University, and the Research Scientist with the Institute of High Performance Computing, Research Agency for Science, Technology, and Research, Singapore. He is currently an Associate Professor with the School of Biomedical Engineering, Southern Medical University, China. His research interests include object detection, object tracking, anomaly detection, medical image analysis and machine learning.

Joey Tianyi Zhou received the Ph.D. degree in computer science from Nanyang Technological University, Singapore, in 2015. He is currently a Scientist with the Institute of High Performance Computing, Research Agency for Science, Technology, and Research, Singapore. He was the recipient of the Best Poster Honorable Mention at ACML 2012, the Best Paper Award from the BeyondLabelerWorkshop on IJCAI 2016, the Best Paper Nomination at ECCV 2016, and the NIPS 2017 Best Reviewer Award. He has was an Associate Editors for IEEE ACCESS and *IET Image Processing*.

Yang Xiao received the B.S., M.S. and Ph.D. degrees from the Huazhong University of Science and Technology, China. He is currently an Associate Professor with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, China. Previously, he was ever the Research Fellow with the School of Computer Engineering and Institute of Media Innovation, Nanyang Technological University, Singapore. His research interests involve computer vision, image processing and machine learning.

Yanan Li received the B.S. degree from Wuhan University,Wuhan, China, in 2012, and the Ph.D. degree from the Huazhong University of science and Technology, Wuhan, China, in 2018. She is currently a Lecturer with the School of Computer Science and Engineering, Wuhan Institute of Technology, Wuhan, China. Her current research interests include computer vision and machine learning, with particular emphasis on image segmentation, domain adaptation, and various computer vision applications in agriculture.

Feng Yang received the M.S. degree in biomedical signal and image processing from Sun Yat- Sen University, China in 1993, and the Ph.D. degree in communication and electronic systems from the South China University of Technology, China in 1998. He joined Division of Image Processing (LKEB) with Leiden University Medical Center, Netherlands, during April 2010 to April 2011, he wasa Visiting Scholar. He is now with the School of Biomedical Engineering, Southern Medical University in China, as a Professor and the Director with the Department of Electronic Technology. His research interests include wavelet analysis, medical image processing and pattern recognition.