AnomalyNet: An Anomaly Detection Network for Video Surveillance

Joey Tianyi Zhou, Jiawei Du, Hongyuan Zhu, Xi Peng, Yong Liu, Rick Siow Mong Goh

Abstract—Sparse coding based anomaly detection has shown promising performance, of which the keys are feature learning, sparse representation, and dictionary learning. In this work, we propose a new neural network for anomaly detection (termed AnomalyNet) by deeply achieving feature learning, sparse representation and dictionary learning in three joint neural processing blocks. Specifically, to learn better features, we design a motion fusion block accompanied by a feature transfer block to enjoy the advantages of eliminating noisy background, capturing motion and alleviating data deficiency. Furthermore, to address some disadvantages (e.g., nonadaptive updating) of existing sparse coding optimizers and embrace the merits of neural network (e.g., parallel computing), we design a novel recurrent neural network to learn sparse representation and dictionary by proposing an adaptive iterative hard-thresholding algorithm (adaptive ISTA) and reformulating the adaptive ISTA as a new long short term memory (LSTM). To the best of our knowledge, this could be *one of first works to bridge the ℓ_1-solver and LSTM* and may provide novel insight in understanding LSTM and model-based optimization (or named differentiable programming), as well as sparse coding based anomaly detection. Extensive experiments show the state-of-the-art performance of our method in the abnormal events detection task.

Index Terms—Video Surveillance, Anomaly detection, Recurrent neural network based sparsity learning.

1 Introduction

With the increasing demand for security, surveillance cameras have been widely deployed as the infrastructure for video analysis. One major challenge faced by surveillance video analysis is detecting abnormal events (see Figure 1 for an intuitive illustration), which requires exhausting human efforts. Fortunately, such a labor-intensive task can be recast as an anomaly detection problem [1], [2], [3] which aims to identify unexpected evens or patterns. Anomaly detection differs from the traditional classification problem in the following aspects: 1) It is very difficult to list all possible negative (anomaly) samples. 2) It is a daunting task to collect sufficient negative samples due to the rarity. To achieve anomaly detection, one of the most popular methods is using the videos of normal events as training data to learn a model, and then detecting the abnormal events which would do not conform the learned model.

Following the aforementioned strategy, sparse coding has successfully applied to anomaly detection [4], [5], which consists of dictionary learning and sparse representation. To be specific, sparse coding based anomaly detection (SCAD) first learns a dictionary from a training data set that only consists of normal events and then discovers the abnormal events that cannot be exactly reconstructed by a few of atoms of the learned dictionary. In other words, SCAS assumes that an abnormal event always leads to a large reconstruction error since it does not appear in the training data. Furthermore, extensive studies [5], [6], [7] have proved that well-established features could remarkably improve the performance of anomaly detection, namely, feature learning and sparse coding have lay onto the heart of SCAD.

- *J. T. Zhou, J. Du,Y. Liu, and R. S. M. Goh are with IHPC, A*STAR; E-mail: {zhouty, dujw, liuyong, gohsm}@ihpc.a-star.edu.sg;*
- *H. Zhu is with I²R, A*STAR; E-mail: zhuh@i2r.a-star.edu.sg;*
- *X. Peng is with College of Computer Science, Sichuan University, Chengdu 610065, China. Email: pengx.gm@gmail.com.*

During past decades, a variety of features have been widely used in SCAD. For example, histogram of oriented gradients (HOG) [8], 3D spatiotemporal gradient [9], and the histogram of oriented flows (HOF) [10] have been extensively used in [6], [11], [12], [13]. The major disadvantage of these works is that the used features are handcrafted while data-driven ones are more favorable since the latter could lead to better performance. To enjoy the representative capacity of neural networks, some recent works tried to marriage deep learning and anomaly detection. For example, [14], [15] proposed a neural network which consists of a recurrent neural network (RNN) accompanied with convolutional filters. Their methods could adaptively learn long range contextual dynamics so that the motion and the appearance are implicitly encoded. Although these methods have shown promising performance, they have suffered from following two limitations. On the one hand, motions and appearances are encoded by the RNN and the convolutional filters separately, which implies that the spatial-temporal relations between motions and appearances are broken. As a result, inferior performance may be achieved. On the other hand, the features are typically learned from scratch without considering the well-established pre-trained model from relevant related tasks. Numerous studies [5], [16] have shown that transferable models could remarkably improve the performance of methods.

To address the above limitations, we propose a new feature learning network which consists of motion fusion block and feature transfer block. Specifically, the motion fusion block compresses video clips into a single image while suppressing the irrelevant background. As a result, the motion and appearance can be simultaneously fused into a single image (See Fig. 6). By feeding the compressed images into the feature transfer block, the spatial-temporal (*i.e.*, appearance and motion) features are extracted based on a transferable model. In other words, we utilize knowledge from other related tasks/domains to boost the performance of feature learning.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2019.2900907, IEEE Transactions on Information Forensics and Security

Fig. 1. Normal events vs. abnormal events. The first and second row corresponds to Avenue and Pedestrian dataset, respectively. From the abnormal frames, we observe that either of motion and appearance is important for anomaly detection. For example, the running and the car appeared in the pedestrian lane are considered as abnormal motion and object respectively since they are not included in the normal events defined by the training data.

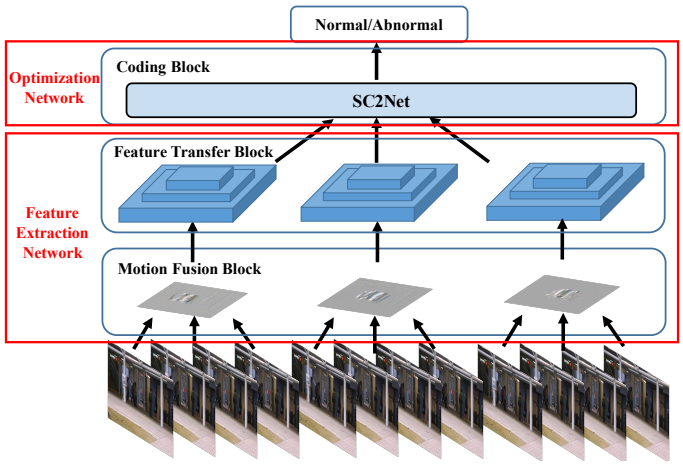


Fig. 2. Conceptual illustration of AnomalyNet.

As one key technique adopted by SCAD, sparse coding requires solving an ℓ_1-regularization optimization accompanied by dictionary learning, which is computationally inefficient, especially in the scenario of video analysis. Although a number of methods have been proposed [17], [18], they have still suffered from following four limitations.

- Most existing ℓ_1-solvers such as the well-known iterative hard-thresholding (ISTA) [19] employ a non-adaptive updating approach by updating the parameters on each dimension with a fixed learning rate. In practice, this strategy may not be optimal in some cases and lead to inferior performance, for example, sparse/big data usually requires the per-dimension updating scheme for saving cost and memory.
- The ℓ_1-solvers such as ISTA do not consider the historical information when designing the updating rules. A lot of studies in the optimization community [20], [21], [22] have proved that incorporating historical information is helpful to improve the convergence performance of algorithms.
- It is very computationally expensive in predicting sparse codes in inference. For each data point, the time complexity for sparse coding is proportional to the size and dimension of the used dictionary, as well as the input dimension.
- Dictionary learning and sparse representation are conducted in an iterative way. In other words, all traditional ℓ_1-solvers may lead to a good sparse representation but will never give a desirable dictionary since sparse representation and dictionary learning are treated separately.

To overcome the first two limitations, we propose a novel ℓ_1-solver, termed adaptive ISTA, by introducing an *adaptive momentum vectors* to enable per-parameter updates and encapsulate the historical information. Accompanying with advantages of the adaptive ISTA, the disadvantage is the difficulty in the optimization of parameters. To be specific, our adaptive ISTA needs automatically learning d_s parameters, whereas ISTA only involves one parameter, where d_s is the dimension of sparse codes. *To further overcome the difficulty in optimization and the last two aforementioned limitations*, we recast our adaptive ISTA as a novel recurrent neural network unit (RNN), termed sparse long short term memory (SLSTM) which could be regarded as a variant of long short term memory (LSTM) [23]. Specifically, the adaptive momentum vectors act as the input and forget gates in the proposed SLSTM. Benefiting from the new formulation, the dictionary and sparse codes are simultaneously optimized, which correspond to the weight and outputs of SLSTM respectively. With the proposed SLSTM unit, we build a neural network (termed SC2Net) to achieve sparse codes in an unsupervised end-to-end manner and use SC2Net as the sparse coding block for anomaly detection.

Unlike the traditional ℓ_1-solvers such as ISTA, SLSTM could address the above four limitations using a novel recurrent neural network. Comparing with LSTM, the proposed SLSTM could perform sparse coding in a different structure. Comparing with existing RNN-based ℓ_1-solvers such as Learned ISTA (LISTA) [24], the major differences are in following aspects. First, SLSTM achieves sparse codes using an LSTM unit instead of a simple RNN unit. In consequence, SLSTM is able to capture historical information which is helpful in speeding up the convergence and improving the performance of our model. In other words, LISTA still suffers from the first two aforementioned limitations like ISTA, whereas our SLSTM does not. Second, the proposed SC2Net does not depend on other sparsity optimizers. In contrast, LISTA requires using the sparse codes given by other ℓ_1-solvers such as ISTA as the supervisor. Such differences make our method working in an end-to-end manner possible.

Based on the aforementioned feature extraction network and optimization network, we propose a new model, termed abnormal event network **AnomalyNet** to detect abnormal events in videos (see Figure 2). The proposed AnomalyNet consists of feature extraction network and the optimization network. To be specific, the feature extraction network consists of the motion fusion block and the feature transfer block. The optimization network is built based on the proposed SLSTM for sparse coding. The major contributions of our work could be summarized as follows:

- To address three challenges in sparse coding based anomaly detection, We propose a novel deep neural network, termed AnomalyNet, which is a unified framework consisting of motion fusion block, feature transfer block and coding block. To be specific, the motion fusion block aims at fusing the appearance and motion information of moving object. The feature transfer block aims to learn a good feature by exploiting the transfer learning ability of deep neural networks, thus alleviating the scarcity of labeled training data. The coding block is a novel neural network which could perform fast inference to achieve sparse coding and thus efficiently detect the abnormal events.
- A novel optimizable network for sparse coding is proposed and applied for anomaly detection. More specifically, we develop a novel variant of ℓ_1-solver by introducing the *adaptive momentum vectors* into the well-known ISTA [19]. The proposed solver (i.e., adaptive ISTA) enables per-parameter updating and encapsulating the historical information into the optimization procedure, thus leading to faster convergence speed and better performance. More interestingly, we unfold the adaptive ISTA as a neural network (i.e., SLSTM) and show it is a variant of the well-known LSTM. To the best of our knowledge, *this could be the first work to bridge the traditional sparse optimization methods and LSTM* and may provide novel insights and understandings in model-based optimization and LSTM.

The paper is a substantial extension of our conference work [25] with further improvements given below. First, we design a new algorithm for anomaly detection by introducing a novel feature extraction network for video surveillance. In contrast, [25] only recasts sparse coding as a neural network (i.e., SC2Net), which does not involve the anomaly detection task. Clearly, it is impossible to detect abnormal events using SC2Net. Second, we present an analysis on SC2Net to explain its effectiveness from the perspective of restricted isometry property (RIP) condition [26], which is important to understanding the working mechanism of our model. Third, the experimental evaluations are totally different. This paper involves new baselines and four abnormal event detection benchmarks, whereas [25] employs SC2Net for image classification only.

2 Related Work

This work mainly involves anomaly detection oriented feature learning, sparse coding (i.e., sparse representation and dictionary learning), and RNN-based optimizers (i.e., LISTA and its variants). In this section, we briefly introduce these three topics one-by-one.

2.1 Feature Learning for Anomaly Detection

Most existing works combine handcrafted features and spatial-temporal information to represent videos for anomaly detection, such as histogram of oriented gradients (HOG) [27], 3D spatiotemporal gradient [28], histogram of oriented tracklets (HOT) [29], and histogram of optical flows [30]. The major disadvantage of these methods is that hand-crafted feature based methods cannot give a desirable performance in complex real-world situations.

To embrace the data-driven feature learning [31], [32], recent attention has shifted from feature engineering to deep neural networks. For example, [33] proposes a two-stream network wherein one stream extracts either appearance or motion. However, the method ignores the connection between appearance and motion, thus breaking the spatiotemporal connection. [3], [34] propose using 3D-CNN to model normal video patterns by partitioning inputs into multiple video cubes. The major challenge is training a 3D-CNN since it involves much more parameters than traditional CNNs. [14] recently proposes ConvLSTM-AE by incorporating convolutional filters into an LSTM to process sequential data in a self-supervised way. However, due to the limitation of architecture, it can only learn features from the local scope and cannot utilize the pre-trained models from other tasks. More recently, more and more researches focus on either of the fully convolutional neural networks (FCNs) [35] and generative adversarial networks (GANs) [36], [37], [38], [39].

These methods have faced the following challenges. On the one hand, they typically learn features from scratch and do not exploit pre-trained features from relevant recognition/detection tasks. Since the data size of anomaly detection is quite small compared with other domains such as ImageNet. Hence, to embrace the merits of neural networks, one of feasible way is to utilize transferable feature/models. On the other hand, it still remains open how to fuse the motion and the appearance to encapsulate the spatiotemporal information into features.

2.2 Sparse Coding for Anomaly Detection

Sparse coding assumes that each sample can be approximately/exactly represented as a linear combination of a few of atoms of a learned dictionary, which has been widely used in anomaly detection [4], [40], [41]. To be specific, sparse coding based anomaly detection learns a dictionary with the sparsity constraint in training and uses the reconstruction loss to identify the irregular frames (i.e., abnormal events) in inference. One major advantage of sparse coding is computational inefficiency, which makes difficulties in real-time applications such as surveillance video analysis. To tackle such a disadvantage, for example, [4] proposes an online sparse coding approach. [40] proposes discarding the sparsity constraint and learning multiple small dictionaries to encode image patches at multiple scales. Besides the computational inefficiency, these existing methods have still suffered from the limitations rooted in ℓ_1-solvers as discussed in Introduction.

Recently, some methods [5] employ LISTA [24] to conduct anomaly detection. Although these methods could enjoy fast inference speed and simultaneously learn dictionary and sparse representation thanks to neural network based implementation, they have suffered from the first two limitations (see Introduction) since they are indeed equivalent to the traditional ℓ_1-solvers. In short, they employ a non-adaptive updating strategy and do not consider the historical information in optimization. Hence, the

obtained sparse representation may be suboptimal and would give an inferior performance in some cases. For example, sparse/big data usually require the per-dimension updating scheme for saving memory and computational source. Furthermore, numerous studies in the optimization community [20], [21], [22] have proved that incorporating historical information is helpful in improving the convergence performance of optimizers.

2.3 LISTAs

LISTA [24] could be one of first works to marriage ℓ_1-solvers and recurrent neural networks, which unfolds ISTA – one of the most popular ℓ_1-solvers, into a simple recurrent neural network. Such a model-based optimization or called differentiable programming has attracted a lot of interests [25], [42], [43], [44], [45], [46], [47], [48] thanks to following advantages. First, the inference speed is very fast since it only progressively passes inputs through a neural network, whereas the traditional ℓ_1-solvers need solving a convex problem. Second, model-based optimizations give a feasible way to intuitively bridge statistical inference methods and neural networks, thus making neural networks interpretable. Specifically, a variety of regularizations (e.g., ℓ_0,) could be reformulated as a layer or activation function [46], [47], [48], [49].

Despite the advantages of LISTA-like methods, LISTA and its variants [42], [50] have faced some challenges. First, LISTAs are actually a supervised method, which uses the precomputed sparse codes of ISTA as supervisors. However, it is a daunting task to obtain supervisors in real-world applications, especially, in the scenario of anomaly detection. Second, the performance of LISTAs is upper bounded by that of ISTA in theory as the former is just an approximation of the latter.

3 Feature Extraction Network

AnomalyNet consists of two subnetworks. In this section, we introduce the first one, i.e., feature extraction network which consists of **Motion Fusion Block (MFB)** and **Feature Transfer Block (FTB)**. In brief, MFB is a dynamic image network [51] which summarizes the appearance and motion of the video sequences. FTB is a well-established neural network which extracts spatiotemporal features from the results given by the motion fusion block.

3.1 Motion Fusion Block

Abnormal events in video data are defined in terms of irregular shapes or motions or both of them. To better reveal the characteristics of motion and appearance, one core task is to adequately capture the dynamic abnormal behavior information. To this end, most of state-of-the-art RGB-based anomaly detection approaches resort to multiple frames input [3], [34], [35] or LSTM architecture [14] or extracting dense optical flow field [37].

Nevertheless, accurate optical flow estimation or 3D convolution architecture is still a challenging task of high computational burden, which is infeasible for the practical applications. Additionally, most existing abnormal event detectors only focus on the image itself, which may face following challenges. First of all, the background may distract the attention in detection and the image may be corrupted by various noises. Hence, ones would eliminate the information irrelevant to image content for better detection. Second, the motion and appearance of objects are both important to anomaly detection. For example, the appearance of a bicycle is different from that of human, and thus the bicycle in the pedestrian lane should be annotated as the anomaly (see Figure 1).

To fully exploit the motion and the appearance of objects, we employ RankSVM [51] to compress the sequence of frames $\phi(\mathbf{I}_t) \in \mathbb{R}^d$ into a single static image $\mathbf{x}$ as follows[1] ,

$$\min_{\mathbf{x}} \frac{\lambda}{2}\|\mathbf{x}\|^2 + \frac{2}{T(T-1)} \sum_{p \geq q} \max\left(0, 1 - S(p|\mathbf{x}) + S(q|\mathbf{x})\right) \quad (1)$$

where $S(q|\mathbf{x}) = \langle \mathbf{x}, \mathbf{v}_t \rangle$ denotes the ranking score associated with the time-step q. $\mathbf{v}_t = \frac{1}{t}\sum_{\tau=1}^{t} \phi(\mathbf{I}_t)$ denotes the average frame within t time-steps. As the optimal feature $\mathbf{x}$ reflects the appearance order of frames, the spatial-temporal dynamic evolution information could be captured. The second term is used to constrain the ranking loss with a unit margin for any $\{q, p\}$, i.e., $\forall\{q, p\}$, if $q > p$, then $S(q|\mathbf{v}_t) > S(p|\mathbf{v}_t)$. For a better illustration, Figure 6 shows the visual comparisons between the output of motion fusion and the raw RGB input.

To efficiently solve Eqn.1, we adopt its first order approximation like [51]:

$$\mathbf{x} \propto \sum_{q>p} \mathbf{v}_q - \mathbf{v}_p = \sum_{t=1}^{T} \beta_t \mathbf{v}_t, \quad (2)$$

where $\beta_t = 2t - T - 1$ is a scalar, which denotes the coefficient. The above approximation makes jointly optimizing RankSVM and CNN possible thanks to the existence of sub-gradient of Eqn.2. In other words, we could recast the RankSVM as a layer to stack onto a CNN, thus fusing the appearance and the motion of moving objects.

The motion fusion block compresses the video or video clips into a single image, while maintaining the rich motion and appearance information. In the context of deep learning (e.g., Convolutional Neural Network (CNN)), it has achieved great success towards RGB-based CNN methods with acceptable computational cost. Compared with other methods for motion characterization, the compressed image takes the advantages over computational efficiency and compactness. Furthermore, using the compressed image can also avoid stacking the sequence of RGB frame as the input of CNN for abnormal behavior description, which could be helpful to prune the complexity of CNN. Note that Deep-Anomaly [35] also considers compressing every two adjacent frames into the averaged frame to capture motion and shape information, which is remarkably different from our method in two aspects. On the one hand, we consider more than just two frames for compression. On the other hand, we take different weights for different frames instead of the simple averaging. In other words, our model generalizes Deep-Anomaly.

3.2 Feature Transfer Block

Existing anomaly detection datasets such as Avenue [1], Pedestrian [7], and Subway [52] are smaller than the datasets from other tasks, e.g., ImageNet for image categorization [53] and Sports-1M [54] for video classification. To address such a small training data issue in anomaly detection, one of the most feasible ways is transfer learning.

In fact, [55] recently conducts experiments to evaluate the effectiveness of transferable representation in anomaly detection. The results show that transferable anomaly detection is a nontrivial

1. $\phi(\mathbf{I}_t)$ denotes the original pixel value or feature representation of frame $\mathbf{I}_t$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2019.2900907, IEEE Transactions on Information Forensics and Security

task and helpful to performance improvement. To utilize the transferable features, we build a feature transfer block using existing pre-trained CNN models for image classification and fine-tune it on a dataset of anomaly detection. This is an important benefit of our method because training large CNNs requires millions of data samples which may be difficult to obtain for video surveillance. In details, the feature transfer block is a residual network [56] with 50 layers (ResNet-50) pre-trained on the ImageNet database. Besides the motion fusion block, we could also extract the deep features directly from the original data by using the pre-trained model to further improve the performance. In the experiments, we will empirically investigate the effectiveness of this block.

4 Optimization Network

The second subnetwork of our AnomalyNet is an optimization network which simultaneously achieves sparse representation and dictionary learning using a novel LSTM network (termed SC2Net). In other words, we conduct sparse coding through optimizing SC2Net.

As SC2Net is a RNN-based implementation of our adaptive ISTA. Thus, we first introduce the adaptive ISTA which is a novel ℓ_1-solver by adding an *adaptive momentum vectors* into the well-known ISTA [19] to overcome the first two limitations. After that, we elaborate on how to reformulate the proposed adaptive ISTA as a neural network (Sparse LSTM, SLSTM) and show its connection with the LSTM. As a result, the last two limitations are overcame through evolving from the algorithm-based optimization to the model-based optimization. In summary, from ISTA to adaptive ISTA to SC2Net, we present an effective way to overcome all four aforementioned limitations suffered by traditional ℓ_1-solvers.

4.1 Adaptive ISTA for Anomaly Detection

After fusing the appearance and the motion of moving objects and obtaining the corresponding features $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T\}$, our model seeks to learn a dictionary $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_{d_s}] \in \mathbb{R}^{d_x \times d_s}$ to exactly/approximately encode all normal events $\mathbf{X}$ using sparse representation $\{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_T\}$. Mathematically, the problem can be formulated as follows,

$$\min_{\mathbf{S},\mathbf{B}} \sum_i \|\mathbf{x}_i - \mathbf{B}\mathbf{s}_i\|_2^2, \;\; s.t. \|\mathbf{s}_i\|_0 \leq k, \;\; \|\mathbf{b}_j\|^2 \leq 1, \;\; j = 1, \cdots k \tag{3}$$

The above optimization is hard to solve due to the non-convexity of ℓ_0-norm. Therefore, the ℓ_0-norm is usually relaxed to the ℓ_1-norm as follows:

$$\min_{\mathbf{S},\mathbf{B}} \sum_i \|\mathbf{x}_i - \mathbf{B}\mathbf{s}_i\|_2^2 + \lambda \|\mathbf{s}_i\|_1^2, \;\; s.t. \|\mathbf{b}_j\|^2 \leq 1, \;\; j = 1, \cdots k \tag{4}$$

To solve Eqn.(4), the most effective way is alternatively optimizing $\mathbf{B}$ or $\mathbf{S}$ while fixing the other, and these two optimization processes correspond to dictionary learning and sparse representation, respectively.

Fixing $\mathbf{S}$, the optimization reduces to the following ℓ_2-constrained optimization problem,

$$\min_{\mathbf{B}} \|\mathbf{X} - \mathbf{B}\mathbf{S}\|_F^2, \;\; s.t. \|\mathbf{b}_i\|^2 \leq 1, \;\; i = 1, \cdots k. \tag{5}$$

Fixing $\mathbf{B}$, the optimization reduces to a sparse approximation problem which aims to represent the input $\mathbf{x}$ as a linear combination of $\mathbf{B}$ by

$$\min_{\mathbf{s}} \sum_i \|\mathbf{x}_i - \mathbf{B}\mathbf{s}_i\|_F^2 + \lambda \|\mathbf{s}_i\|_1. \tag{6}$$

The iterative hard-thresholding (ISTA) algorithm is one of the most effective optimizers to solve Eqn.(6). It decomposes the objective of Eqn.(6) into two parts. Namely, the differentiable part $g(\mathbf{s}) = \|\mathbf{x} - \mathbf{B}\mathbf{s}\|_F^2$ is updated by the gradient descent and ℓ_1 part is updated by the hard thresholding operator. The updating formula can be mathematically expressed as follows,

$$\mathbf{s}^{(t)} = sh_{(\lambda\tau)}(\mathbf{s}^{(t-1)} - \tau \nabla g(\mathbf{s}^{(t-1)})), \tag{7}$$

where the shrinkage function is defined as $sh_{(\lambda\tau)}(\mathbf{s}) = \text{sign}(\mathbf{s})(|\mathbf{s}| - \lambda\tau)_+$. Then, the solution of Eqn.(7) can be achieved via the following updating rule,

$$\begin{aligned} \mathbf{s}^{(t)} &= sh_{(\lambda\tau)}(\mathbf{s}^{(t-1)} - \tau(\mathbf{B}^\top(\mathbf{B}\mathbf{s}^{(t-1)} - \mathbf{X}))) &(8)\\ &= sh_{(\lambda\tau)}(\mathbf{W}_e\mathbf{s}^{(t-1)} + \mathbf{W}_d\mathbf{x}), &(9)\end{aligned}$$

where $\mathbf{W}_e = \mathbf{I} - \tau\mathbf{B}^\top\mathbf{B}$, $\mathbf{W}_d = \tau\mathbf{B}^\top$.

Despite the success of ISTA, it suffers from following limitations: 1) ISTA is a non-adaptive updating approach, which updates the parameters on each dimension with a fixed learning rate. Clearly, such a strategy may lead to inferior performance; 2) ISTA does not utilize the historical information for updating. In contrast, the historical information has shown promising performance in speeding up the convergence performance.

To solve the aforementioned problems, we propose a novel ℓ_1-solver by introducing an *adaptive momentum vectors* into ISTA motivated by recent development in the community of optimization. To be specific, a number of algorithms have been proposed to optimize neural networks by incorporating the "momentum" into the dynamics of stochastic gradient descent (SGD). These methods have shown promising performance in improving the robustness and convergence speed of SGD since the momentum incorporates the historical updating information [20]. To further improve the performance of the momentum-based SGD, Adagrad [22] and AdaDelta [21] introduce adaptation into SGD so that the learning rate varies with parameters. The basic idea behind them is performing larger updates for infrequent parameters and smaller updates for frequent parameters. Extensive numerical studies have demonstrated that the adaptation drastically improves convergence performance over the non-adaptive SGD methods.

Borrowing the high-level idea of these optimization methods, we introduce *adaptive momentum vectors* $\mathbf{i}^{(t)}, \mathbf{f}^{(t)}$ into ISTA at the time step t as follows,

$$\begin{aligned} \tilde{\mathbf{c}}^{(t)} &= \mathbf{W}_e\mathbf{s}^{(t-1)} + \mathbf{W}_d\mathbf{x} \\ \mathbf{c}^{(t)} &= \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)} \\ \mathbf{s}^{(t)} &= sh_{(\lambda\tau)}(\mathbf{c}^{(t)}), \end{aligned} \tag{10}$$

where $\odot$ is the element-wise product of the vectors. Following the above notations, the updating rule in ISTA can be equivalently expressed to $\mathbf{s}^{(t)} = sh_{(\lambda\tau)}(\tilde{\mathbf{c}}^{(t)})$.

Different from ISTA, our method considers the role of not only the current information but also the previous information. More specifically, it formulates the linear combination of $\mathbf{c}^{(t-1)}$ at the previous iteration and $\tilde{\mathbf{c}}^{(t)}$ at the current iteration which are weighted by adaptive momentum vectors $\mathbf{f}^{(t)}$ and $\mathbf{i}^{(t)}$, respectively. The adaptive momentum vectors allow the combination of two outputs at the level of parameters, which is different from directly applying momentum methods into ISTA. We further pass $\tilde{\mathbf{c}}^{(t)}$ into the shrinkage function again to ensure the sparsity. We name this method as **adaptive ISTA** in the upcoming sections. In the adaptive ISTA, $\mathbf{c}^{(t)}$ accumulates all the historical information

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2019.2900907, IEEE Transactions on Information Forensics and Security

with different weights $\mathbf{f}^{(t)}, \mathbf{i}^{(t)}$ for the iteration t, which is spiritually similar to the diagonal matrix containing the sum of the squares of the past gradients in Adagrad.

4.2 Sparse Long Short Term Memory Unit (SLSTM)

Besides the last two limitations faced by almost all ℓ_1-solvers, our adaptive ISTA overcomes the difficulty in parameter learning, namely, how to adaptively determine the values of momentum vectors $\mathbf{f}^{(t)}, \mathbf{i}^{(t)}$. Existing SGD methods such as AdaDelta solve a similar problem by empirically reducing the value of momentum after fixed training epochs. However, such a strategy is unsuitable to our case since the momentum in our adaptive ISTA is a vector instead of a constant. Thus, it is preferable to learn $\mathbf{f}^{(t)}$ and $\mathbf{i}^{(t)}$ from data.

To achieve the above end, we propose parameterizing the adaptive momentum vectors with the output of sparse codes at the previous layer as well as input data such that $\mathbf{f}^{(t)}$ and $\mathbf{i}^{(t)}$ are learned from data without tedious hand-craft tuning. More interestingly, such an idea could be implemented by recasting the adaptive ISTA as a novel LSTM unit. The unit is termed as sparse LSTM (SLSTM, see Figure 3) wherein "input gate" and "forget gate" correspond to $\mathbf{i}^{(t)}$ and $\mathbf{f}^{(t)}$ respectively. Noticed that, SLSTM does not have "output gate" like the vanilla LSTM. The SLSTM unit is achieved by rewriting (10) as follows:

$$\mathbf{i}^{(t)} = \sigma(\mathbf{W}_{is}\mathbf{s}^{(t-1)} + \mathbf{W}_{ix}\mathbf{x}), \quad (11)$$

$$\mathbf{f}^{(t)} = \sigma(\mathbf{W}_{fs}\mathbf{s}^{(t-1)} + \mathbf{W}_{fx}\mathbf{x}), \quad (12)$$

$$\tilde{\mathbf{c}}^{(t)} = \mathbf{W}_{e}\mathbf{s}^{(t-1)} + \mathbf{W}_{d}\mathbf{x}, \quad (13)$$

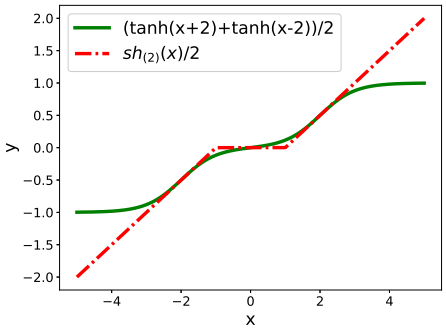
$$\mathbf{c}^{(t)} = \mathbf{f}^{(t)} \odot \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \odot \tilde{\mathbf{c}}^{(t)}, \quad (14)$$

$$\mathbf{s}^{(t)} = h_{(\mathbf{D},\mathbf{u})}(\mathbf{c}^{(t)}), \quad (15)$$

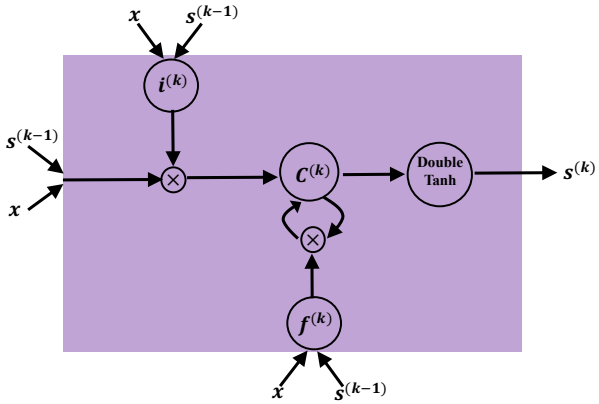
where $\mathbf{W}$ denotes the weight matrix (e.g., $\mathbf{W}_{is}$ is the weight matrix from the input gate to the outputs), $\sigma(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{x}}}$, $h_{(\mathbf{D},\mathbf{u})} = \mathbf{D}(tanh(\mathbf{x}+\mathbf{u}) + tanh(\mathbf{x}-\mathbf{u}))$. $\mathbf{u}$, $\mathbf{D}$ denote a trainable vector and diagonal matrix, respectively.

ISTA and LISTA exclusively obtain sparse representation based on the previous output. This kind of architecture leads to the so-called "error propagation phenomenon". In details, the error in the first few layers will be propagated and further amplified in the upcoming layers. Furthermore, once the useful information is discarded by the previous layers, the upcoming layers will have no chance to utilize the discarded information. Fortunately, this issue can be alleviated with the use of "cell" state $\mathbf{C}^{(t)}$ in our SLSTM. The "cell" plays as another "eye" to supervise the optimization, thus giving two major advantages. First of all, it captures long-term dependence from the previous outputs. In addition, it automatically accumulates important information and forgets useless or redundant information in the dynamics of neural networks.

It is worth noting that we use smooth and differentiable nonlinear activation function named "*Double tanh*" instead of the shrinkage function for following two reasons. On the one hand, the cell recurrent connection needs a function whose the second derivative sustains for a long span to address the vanishing gradient problem [57]. On the other hand, the *Double tanh* function could approximate to the shrinkage function well within the interval of $[-\mathbf{u}, \mathbf{u}]$ [24]. Figure 3(a) gives a comparison between *Double tanh* and shrinkage functions.



(a) Examples of the Double tanh (in green color) and Shrinkage (in red color) functions.



(b) Sparse LSTM Unit (SLSTM Unit).

Fig. 3. Sparse LSTM Network (SLSTM).

4.3 Sparse Coding to Network (SC2Net)

LISTA needs using the traditional ℓ_1-solver (e.g., ISTA) to precompute $\mathbf{s}^*$ as the supervisor, which leads to a high computational cost. In other words, the performance of LISTA largely depends on the quality of $\mathbf{s}^*$. To overcome the drawback, we propose a novel optimization framework based on SLSTM, termed SC2Net. Specifically, the sparsity loss and the reconstruction loss are incorporated into our SC2Net to supervise the optimization process. With the sparsity loss, SC2Net could give sparse codes in parallel. With the reconstruction error, SC2Net is no longer an approximation to existing SC methods. In other words, it does not require computing $\mathbf{s}^*$ in advance.

For any data point $\mathbf{x}$, we propose the following reconstruction loss:

$$\|\mathbf{x} - \frac{1}{\tau}\mathbf{W}_d^\top \mathbf{s}\|_F^2 \quad (16)$$

where $\mathbf{s}$ is the output of encoding part in the network w.r.t. $\mathbf{x}$ and $\mathbf{B} = \frac{1}{\tau}\mathbf{W}_d^\top$ (Eqn. 9). Here, we do not learn an individual decoding matrix. Instead, we reuse the encoding matrix $\mathbf{W}_d$. Such a strategy gives two advantages: 1) it maintains the physical meaning of the original formulation (Eqn. 9), i.e., the encoding matrix is the transpose of the decoding matrix; 2) it reduces the computation cost to train our model.

To further enhance the sparsity of the solution, the ℓ_1 loss is also considered in our formulation. The overall cost function for

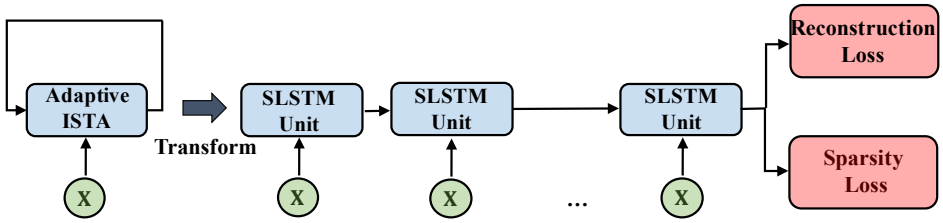


Fig. 4. The Architecture of SC2Net.

SC2Net is defined as follows, [2],

$$\|\mathbf{x} - \frac{1}{\tau}\mathbf{W}_d^\top \mathbf{s}\|_F^2 + \lambda\|\mathbf{s}\|_1, \tag{17}$$

The architecture of SC2Net is illustrated in Figure 4.

The advantages of recasting the sparse coding as a neural network are in three-fold. 1) It facilitates an end-to-end training and does not elaborately choose hyper-parameters for optimization. In other words, the hyper-parameters are learned from data; 2) Comparing with the standard neural networks, SLSTM is with high interpretability instead of a "black-box" since it is an implementation of an ℓ_1-optimizer. Namely, ones could know the physical/mathematical meanings of each layer, middle output, and so on; 3) Comparing with popular ℓ_1-solvers, SLSTM boosts the inference speed since it simply passes the input through the network and avoids solving an ℓ_1 optimization problem in inference. Furthermore, although SLSTM is induced from the proposed adaptive ISTA, it simultaneously learns sparse representation and dictionary, whereas traditional methods including adaptive ISTA do not update the dictionary.

4.4 Analysis on SLSTM

Although neural network based optimizations have shown promising performances in numerous experimental studies [44], [46], only few of works provide theoretical explanations towards their success. In this section, we conduct analysis on the proposed SLSTM by utilizing the well-known Restricted Isometry Property (RIP) [26] which is one of the most important properties to guarantee the sparsity. In the following, we will first introduce some preliminaries about RIP and then experimentally show why the proposed SLSTM could achieve better performance. More specifically, we employ the mutual coherence induced by the RIP to measure the orthogonality of the dictionary learned by SLSTM. A higher value indicates a better dictionary, larger orthogonality, and higher sparsity.

4.4.1 Preliminaries of RIP

Our analysis is based on the well-known RIP which is briefly introduced as below.

Definition 1 (Restricted Isometry Property [58]). A matrix $\mathbf{B}$ is said to satisfy the k-restricted isometry property (RIP) with constant $\delta_k[\mathbf{B}] < 1$ if

$$(1 - \delta_k[\mathbf{B}])\|\mathbf{s}\|_2^2 \leq \|\mathbf{B}\mathbf{s}\|_2^2 \leq (1 + \delta_k[\mathbf{B}])\|\mathbf{s}\|_2^2 \tag{18}$$

holds for all $\{\mathbf{s} : \|\mathbf{s}\|_0 \leq k\}$, where k measures the sparsity. The Restricted Isometry Constant (RIC) is the smallest value $\delta_k[\mathbf{B}]$ satisfying the above equation.

According to [58], [59], one could obtain:

Lemma 1 (Uniqueness). Assuming $\mathbf{B}$ satisfies RIP of order ck with the constant $\delta_{ck}[\mathbf{B}] < \kappa_{ck}$, where $c = \{1, 2, 3, 4\}$. There exists an optimizer $\mathbf{s}^*$ to the problem

$$\mathbf{B}\mathbf{s} = \mathbf{x} \quad s.t. \quad \|\mathbf{s}\|_0 \leq k. \tag{19}$$

In other words, $\mathbf{s}$ is the unique minimizer w.r.t. both ℓ_0- and ℓ_1-norms.

Lemma 1 states that with the proper RIC, the optimal solution to (19) can exactly recover the k-sparse signals.

The RIP implies that the smaller RIP constant $\delta_s[\mathbf{B}]$, the lower correlation among sub-matrices of $\mathbf{B}$ with s columns. However, it is NP-hard to verify the RIP condition for a matrix [60]. In practice, it is more feasible to use the mutual coherence of a matrix to judge the RIP condition with the following definition,

Definition 2 (Coherence [61]). The mutual coherence of $\mathbf{B}$ is defined by

$$\mu(\mathbf{B}) = \max_{k,j,k\neq j} \frac{|\mathbf{b}_k{}^\top \mathbf{b}_j|}{\|\mathbf{b}_k\|_2\|\mathbf{b}_j\|_2}, \tag{20}$$

where $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_n]$.

The mutual coherence reflects how well spread the directions of a collection of vectors are, namely, the lower mutual coherence of $\mu[\mathbf{B}]$, a greater spread of directions and lower coherence of the dictionary $\mathbf{B}$. The mutual coherence and RIC can be mutually expressed as shown in the following lemma.

Lemma 2 ([62]). Without loss of generality, let the column vectors $\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_n$ of $\mathbf{B}$ be normalized, then $\mathbf{B}$ satisfies the RIP of order k with parameter $\delta_k[\mathbf{B}] = (k-1)\mu[\mathbf{B}]$.

Remark 1. Lemma 2 shows that the coherence can be replaced by RIC to measure the possibility of sparse signal recovery, which gives a large convenience in practice since the former is easier to compute. Extensive studies show that the smaller coherence of the dictionary, the more easily sparse signals can be recovered. Therefore, in the following, we will adopt the mutual coherence to measure the orthogonality degree of dictionary and show that neural network based optimization approaches can learn the dictionary with the smaller coherence $\mu(\mathbf{B})$ compared to the alternating optimization based algorithms. In short, we propose using the mutual coherence to evaluate the performance of neural network based sparse coding.

2. In the experiments, the network is often learned through minimizing the average cost over a set of training samples using a stochastic gradient method.

4.4.2 Why SLSTM Makes Sense?

To show the effectiveness of SLSTM, we conduct experimental analysis on synthetic data sets. To be specific, we generate 10k vectors $\mathbf{s} \in \mathbb{R}^{d_s}$ as the ground truth and each vector includes ρ randomly selected nonzero entries. The value of nonzero entries follow the uniform distribution $\mathbf{U}[-0.7, 0.7]$ excluding the interval $[-0.3, 0.3]$ to avoid small and relatively inconsequential contributions to the sparsity support. Moreover, We obtain inputs $\mathbf{x} \in \mathbb{R}^{d_x}$ via $\mathbf{x} = \mathbf{B}\mathbf{s}$, where two different dictionaries are considered:

1) Assumption 1 (Low-rank Dictionary): In the experiment, we design a coherent dictionary matrix $\mathbf{B} = \eta\mathbf{R} + \mathbf{K}_r$, where $\mathbf{K}_r = \mathbf{U}\mathbf{V}$ and $\mathbf{U} \in \mathbb{R}^{d_x \times r}, \mathbf{V} \in \mathbb{R}^{r \times d_s}$. In addition, $\mathbf{U}$ and $\mathbf{V}$ have i.i.d. elements drawn from $U(0, 1)$. We set $\eta = 0.3, r = 5, d_x = 20, d_s = 100$. The result is reported in Table 1.
2) Assumption 2 (Clustered Dictionary): We construct a coherent dictionary matrix $\mathbf{B} = [\mathbf{B}_1, \mathbf{B}_2, \cdots, \mathbf{B}_C]$ with $\mathbf{B}_j = \mathbf{u}_j\mathbf{d}_j^\top + \eta\mathbf{R}_j$, where $\mathbf{u}_j \in \mathbb{R}^{d_x}, \mathbf{d}_j \in \mathbb{R}^{d_j}$ have i.i.d. elements drawn from $U(0, 1)$. We set $\eta = 0.3, C = 50, d_x = 20, d_s = 100, d = d_s/C = 2$. The result is reported in Table 2.

TABLE 1
Assumption 1: Mutual Coherence Comparisons. The lower mutual coherence, the better dictionary.

ρ	Original Dict	LISTA	LFISTA	SLSTM
$\rho = 2$	0.9895	0.7956	0.7253	**0.5815**
$\rho = 4$	0.9972	0.8500	0.8059	**0.6087**
$\rho = 6$	0.9937	0.8033	0.8893	**0.7244**
$\rho = 8$	0.9962	0.8003	0.8156	**0.6232**

TABLE 2
Assumption 2: Mutual Coherence Comparisons. The Lower mutual coherence, the better dictionary.

ρ	Original Dict	LISTA	LFISTA	SLSTM
$\rho = 2$	0.9984	0.8867	0.7739	**0.2401**
$\rho = 4$	0.9895	0.8156	0.8078	**0.2105**
$\rho = 6$	0.9972	0.8314	0.8441	**0.2201**
$\rho = 8$	0.9960	0.8213	0.8121	**0.1893**

For fair comparisons, we compare the proposed SLSTM with LISTA [24] and LFISTA [50] with the same loss defined in Eqn (17). From Tables 1–2, one could observe that in both two cases, neural network based optimizations (LISTA, LFISTA, and SLSTM) learn a better dictionary in terms of the mutual coherence. Interestingly, in a more complicated case (Assumption 2), SLSTM achieves a more significant advantage in reducing the mutual coherence compared with the simple RNN based optimization methods (i.e., LISTA and LFISTA). This verifies our basic idea, namely, historical information is helpful in improving the performance of sparse coding, which is encapsulated into our SLSTM.

5 Experiments on Abnormal Event Detection

In this section, we investigate the performance of AnomalyNet on the task of abnormal event detection using two real-world datasets.

5.1 Datasets

We carry out experiments on two benchmark datasets widely used for anomaly detection, namely, CUHK avenue [1], UCSD Pedestrian [7] and UMN [63]. The training and testing data are split by following the default setting. Some samples from these datasets are illustrated in Figure 5.

- CUHK Avenue dataset contains 30,652 frames which are partitioned into 16 training and 21 testing video clips. In the testing video clips, 47 abnormal events are contained, which are either 1) the circulation of nonpedestrian entities in the walkways, or 2) anomalous pedestrian motion patterns. Commonly occurring anomalies include bikers, skaters, small carts, and people walking across a walkway or in the surrounding grass. A few instances of wheelchairs are also recorded. All abnormalities occur naturally, i.e., they are not staged or synthesized for data collection.
- UCSD Pedestrian dataset [7] is acquired with a stationary camera mounted at an elevation and pedestrian walkways, which includes two subsets. Namely, Ped1 and Ped2 which contain 7,200 frames with 40 abnormal events and 2,010 frames with 12 abnormal events, respectively. Videos are from the outdoor scene, recorded with a static camera at 10 fps. All other objects except for pedestrians are considered as irregularities.
- UMN Dataset [63] consists of normal and abnormal crowd videos which are collected on the University of Minnesota. The dataset comprises three different scenarios of an escape event in different indoor and outdoor scenes. In each scenario, a group of people normally walks in an area, but suddenly all people run away (escape). In other words, the escape is considered to be the anomaly.

5.2 Evaluation

In the training phase, we learn a dictionary $\mathbf{B}$ to encode the normal events by minimizing the reconstruction error. During testing, such a dictionary cannot exactly/approximately reconstruct the abnormal patch $\mathbf{x}_t$ which would assume leading to a large reconstruction error, i.e., $l(t) = \|\mathbf{x}_t - \mathbf{B}\mathbf{s}_t\|_2^2$. Regarding different scales, we use the average reconstruction error of all patches within the current frame to represent the reconstruction error of all frames. Different from most existing methods, AnomalyNet requires passing L frames to construct dynamic images in the motion fusion block, where L is fixed through training to inference. For fair comparisons, we normalize the errors into range $[0, 1]$ and further calculate the regularity score via,

$$s(t) = 1 - \frac{l(t) - \min_t l(t)}{\max_t l(t) - \min_t l(t)}. \tag{21}$$

During the testing phase, our method marks the testing images abnormal if more than half of frames are anomalies before compression.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2019.2900907, IEEE Transactions on Information Forensics and Security

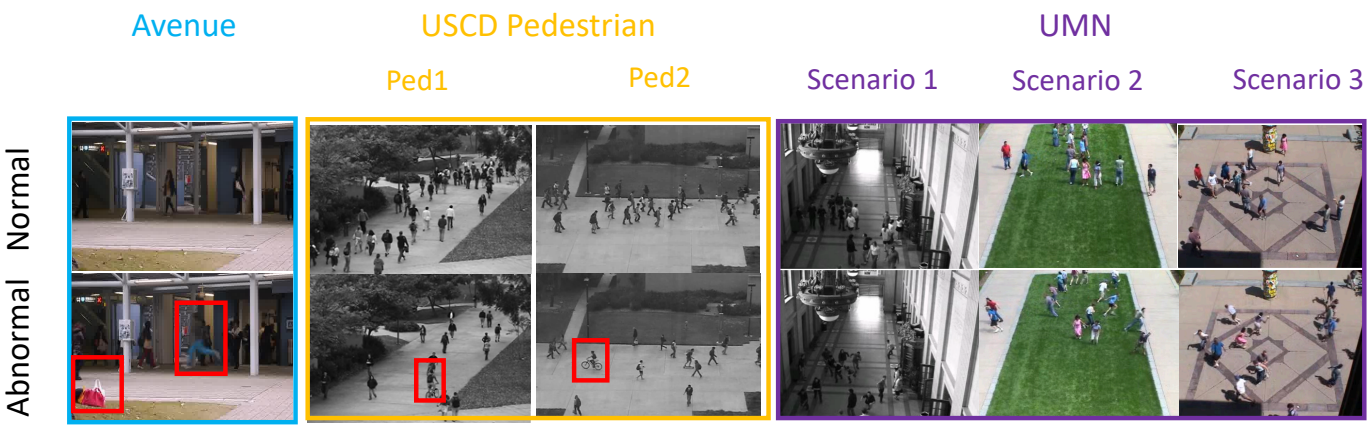


Fig. 5. Some normal and abnormal frames sampled from the CUHK Avenue, UCSD Pedestrian and UMN datasets. Red boxes denote anomalies in abnormal frames. Different from the Avenue and Pedestrian dataset, all the people in the abnormal frame are considered as anomalies in UMN dataset.

5.3 Evaluation Metric

The regularity score can be used to judge whether the input frame is normal or abnormal. The threshold of regularity score is used to identify the abnormal frames, which is manually specified. The optimal value of this parameter is very important since a higher threshold leads to a higher false negative rate, while a lower threshold leads to a higher false negative. Thus, the Area Under Curve (AUC) is a more suitable metric [5], [7], which measures the performance by changing different thresholds. In addition, we also evaluate the performance using the equal error rate (EER), which is used in [35], [39]. For a more comprehensive comparison, we also list the results with precision, recall positive, true positive, and false alarm.

We adopt two scales of measurement, i.e. at frame level and at pixel level. More specifically,

- Frame-Level: An algorithm predicts which frames contain anomalous events. This is compared to the clipped frame-level ground-truth anomaly annotations to determine the number of true- and false-positive frames.
- Pixel-Level: An algorithm predicts which pixels are related to anomalous events. This is compared to the pixel-level ground-truth anomaly annotation to determine the number of true-positive and false-positive frames. A frame is a true positive if 1) it is positive and 2) at least 40 percent of its anomalous pixels are identified. Meanwhile, a frame is a false positive if it is negative and any of its pixels in all the scales are predicated as anomalous.

Note that we use the both frame-level and pixel-level for AUC and frame-level for EER.

5.4 Implementation Details

We employ the default AdaDelta [21] optimizer to train AnomalyNet with a GPU of NVIDIA TITAN X in TensorFlow. The batch-size is fixed as 210 and the max training epoch is set to 50. The implementation details of each network are elaborated as below:

- **Motion Fusion Block:** We follow the setting of [51] to adopt the square rooting kernel maps $\sqrt{\cdot}$ and time-varying mean vectors. Dynamic images for each color channel of RGB images are separately generated and then merged so that they can be directly input to the upcoming feature transfer block. We conduct ℓ_2-normalization and scale layers to constrain the range of outputs into $[0, 255]$. We empirically found that the performance could keep stable when the window size T of motion fusion ranges into $[10, 30]$. For simplicity, we set the length of each video clip (i.e., T) to 20 in all experiments. In addition, we could also extract features directly from RGB images to boost up performance.
- **Feature Transfer Block:** To utilize existing CNN networks for learning spatial feature representation, we extract features from the last identity block (before average pooling layer) which is a resnet-50 pretrained on the ImagNet dataset. As the feature map size $7 \times 7 \times 2048$ is quite large for our coding block, we divide each 7×7 feature map into a 1×1,2×2, and 4×4 subregion; and then apply max-pooling over each subregion to finally get a 2048-dim feature. During this process, each original feature is sampled three times, thus generating $1 \times 1 + 2 \times 2 + 4 \times 4 = 21$ samples. For the missing regions, zero padding is applied.
- **Coding Block:** The input dimension of the coding block is fixed as 2048. In implementations, we experimentally found that all evaluated methods perform stable when λ ranges between 0.01 and 1, and $\lambda = 0.1$ usually leads to the best performance. Thus, we fix the sparsity parameter $\lambda = 0.1$ for all methods. [5] empirically show that changing dictionary size can further improve performance. However, such an operation requires extensive manually tuning on different datasets. For simplicity and fair comparisons, we fix the dictionary size as 2048×2048. In other words, all evaluated methods including the proposed one use the same objective function with the fixed sparsity regularization parameter λ. The only one difference among them is the choice in the optimizer.

5.5 Results on the CUHK Avenue Dataset

We first compare our abnormal behavior detection framework with seven state-of-the-art deep learning approaches [1], [3], [5], [14], [37], [64], [65] on the Avenue dataset and various metrics in Table 3. One could observe that generative adversarial network

TABLE 3
Comparisons with state-of-the-art methods on the Avenue dataset which consists of 47 abnormal events in testing.

Methods	Frame AUC	Pixel AUC	ERR	Precision/Recall	True Positive/False Alarm
Lu *et al.* [1]	80.9%	92.9%	-	92.3%/ -	12/1 (old dataset)
3DConv-AE [3]	70.2%	-	25.1%	91.8%/95.7%	45/4
ConvLSTM-AE [64]	77.0%	-	-	-	-
ConvLSTM [14]	-	-	-	95.2%/**92.3%**	40/2
Smeureanu *et al.* [65]	84.6%	93.5%	-	-	-
SRNN [5]	81.7%	-	-	-	-
Liu *et al.* [37]	84.9%	-	-	-	-
AnomalyNet	**86.1%**	**94.1%**	**22.0%**	**95.6%**/91.6%	43/2

TABLE 4
Comparisons with state-of-the-art methods on the Ped1 dataset which consists of 40 abnormal events in testing.

Methods	Frame AUC	Pixel AUC	ERR	Precision/Recall	True Positive/False Alarm
MPPCA [7]	59.0%	20.5%	35.6%	-	-
HOFME [66]	68.8%	21.3%	33.3%	-	-
3DConv-AE [3]	81.0%	-	27.9%	86.4%/95.0%	38/6
ConvLSTM-AE [64]	75.5%	-	-	-	-
ConvLSTM [14]	-	-	27.9%	85.1%/100%	40/7
Liu *et al.* [37]	83.1%	33.4%	23.5%	83.7%/90%	36/7
Deep-Cascade [34]			**9.1%**		
AVID [39]	-	-	12.3%	-	
AnomalyNet	**83.5%**	**45.2%**	25.2%	**88.9%/100%**	**40/5**

TABLE 5
Comparisons with state-of-the-art methods on the Ped2 dataset which consists of 12 abnormal events in testing.

Methods	Frame AUC	Pixel AUC	ERR	Precision/Recall	True Positive/False Alarm
MPPCA [7]	69.3%	-	30%	-	-
MPPC+SFA [7]	61.3%	-	36%	-	-
HOFME [66]	89.9%	-	19.0%	-	-
3DConv-AE [3]	90.0%	-	21.7 %	92.3% / 100%	12/1
ConvLSTM [14]	-	-	-	92.3% / 100%	12/1
GANs [67]	93.5%	-	14%	-	
AMDN [68]	90.8%	-	17.0%		
Deep-Cascade [34]	-	-	**8.2%**	-	
AVID [39]	-	-	14%	-	
SRNN [5]	81.7%	44.8%	-	-	-
Liu *et al.* [37]	**95.4%**	40.6%	12%	91.7%/91.7%	11/1
ALOCC [69]	-	-	13%	-	
Deep-Anomaly [35]	-	-	11%	-	
AnomalyNet	94.9%	**52.8%**	10.3%	**92.3% / 100%**	**12/1**

based method [37] gives a significant improvement over other baselines. Nevertheless, our AnomalyNet still outperforms the other baselines including the GAN-based approach by a large margin in terms of AUC. Moreover, Avenue dataset consists of 47 different events in total. Compared to the other baselines, the proposed method is able to increase the number of detected anomalies without increasing false alarms. However, it fails to detect several abnormal events of jogging that occur in the background where most of the "normal" walking takes place. Since the deviation in regularity caused by jogging in the background is less significant than larger or more disruptive abnormal events like standing on the grass, the evaluation algorithm is unable to distinct the action of jogging from walking pedestrians.

We also visualize the regularity score with varying frames on the Avenue dataset in Figure 7 from which, one could observe that low regularity scores correspond to abnormal events and high scores correspond to normal events.

5.6 Results on the UCSD Pedestrian Dataset

UCSD Pedestrian dataset consists of two different data sets, i.e., Ped1 and Ped2. Ped1 contains a variety of abnormal events that can be classified into two main categories, i.e., the movement

TABLE 6
Comparisons with state-of-the-art methods on the UMN Dataset.

Methods	SF [70]	H-MDT CRF [71]	AVID [39]	GANs [67]	Deep-Cascade [34]	AnomalyNet
EER	12.6 %	3.7%	2.6 %	-	2.5%	2.6 %
AUC	94.9 %	99.5 %	99.6 %	99.0 %	99.6%	99.6%

of non-pedestrian entities and anomalous pedestrian motions. Comparing with Ped1, Ped 2 features a different walkway, which contains fewer anomalies. We compare our method with some state-of-the-art approaches including two handcrafted features based methods [7], [66] and six deep learning based methods [3], [5], [14], [34], [37], [39], [64] on these two datasets. The results for Ped1 and Ped2 are summarized in Table 4 and 5, respectively. In experiments, we conduct comprehensive metric evaluations on Ped1 and Ped2. From the results, one could see that most methods usually perform better on Ped2. The possible reason is that Ped2 is simpler than Ped1 and its variance of crowd density is much smaller than that of Ped1.

5.7 Results on the UMN Dataset

Different from the above used benchmark datasets, the UMN data set 1) does not includes pixel-level ground truth and the anomalies are staged. Furthermore, it produces very salient changes in the average motion intensity of the scene. Therefore, we just follow the common experiment setting in [34], [39], [67] to conduct experiments on three individual scenes and report the average frame-level AUC and EER score for all the three scenarios in Table 6. In this evaluation, our method is compared with several recently-proposed approaches [34], [39], [67], [70], [71], which achieves the best performance in general. Note that, all the methods achieved very high performance on this dataset since all the people in the abnormal frames are considered anomalies making this dataset less challenging. In addition, abnormal cases produce very salient changes in the average motion intensity of the scene, as shown in Figure 6.

TABLE 7
Influence of Different Blocks

Methods	AUC	ERR
SC2Net+FTB	79.4%	25.3%
SC2Net+MFB	77.6%	26.2%
ISTA+MFB+FTB	83.0%	24.5%
LISTA+MFB+FTB	84.3%	23.8%
AnomalyNet	**86.1%**	**22.0%**

5.8 The Influence of Different Blocks

To investigate the individual contribution of these three blocks, we conduct a series of ablative studies on the Avenue dataset in this section.

5.8.1 Motion Fusion Block

Motion fusion block reorganizes and compresses the original RGB raw pixels of a sequence of frames into a single image. To verify the effectiveness of this block for anomaly detection task, we develop a baseline by replacing motion pooling block with the original raw pixels input, termed **SC2Net+FTB**. The empirical comparison is summarized in Table 7 which shows that given the same raw RGB input, our method remarkably outperforms all the baselines including deep learning based methods. The result demonstrates the superiority of the remaining two blocks. Moreover, Figure 6 gives a visualization of output of motion fusion block. From the result, one could observe that the output of motion pooling block actually encodes the dynamic evolution information from all the frames. Spatially reordering features from 1D to 2D can construct a dynamic image for video representation. In addition, the dynamic motion information within the video frames can be revealed by one single dynamic image, while suppressing background as shown in Figure 6. The motion temporal order is also reflected by the gray-scale value. Comparing with the output of motion fusion block, the single original RGB images barely reveal the motion information from every single frame.

TABLE 8
Influence of Different Pre-pretrained Models

Models	AUC	ERR
Alexnet	70.0%	35.1%
VGG-16	84.7%	22.7%
Resnet-50	86.1%	22.0%
Resnet-152	**87.5%**	**21.8%**

TABLE 9
Parameter Analysis: T denotes the frames to be compressed in the motion fusion block and λ is the sparsity parameter used in the optimization network.

λ	AUC/ERR	T	AUC/ERR
0.001	82.1%/26.1%	5	77.8%/30.3%
0.01	80.6%/27.0%	10	81.0%/27.1%
0.1	**86.1%/22.0%**	20	**86.1%/22.0%**
0.5	84.2%/24.3%	40	83.4%/23.9%

5.8.2 Feature Transfer Block

To illustrate the effectiveness of the feature transfer block, we design a baseline **SC2Net+MFB** by removing the feature transfer block from the proposed network. The corresponding experimental results are summarized in Table 7 which shows that transferring knowledge from other relevant tasks could boost the performance by around 6%, especially when the dataset is small. Moreover, the proposed network also outperforms ConLSTM-AE [14] and LSTM-AE [72] with the same pre-trained model by a large performance margin. Furthermore, we conduct experiments by replacing feature transfer block with different pre-trained models

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2019.2900907, IEEE Transactions on Information Forensics and Security

Fig. 6. RGB input vs. Output of Motion Fusion: the motion fusion block shows advantages of removing the irrelevant background and focusing on the abnormal objects. Note that, in the second example, it is hard to visually detect the bicycle (anomaly) from the front view, however our motion fusion block is able to visually keep the moving pattern information with a black tail in a single compressed image.

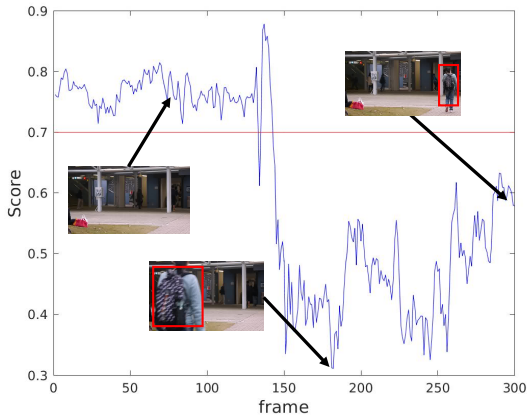


Fig. 7. Regularity Scores v.s. Frames

including Alexnet, VGG-16, Resnet-50, and Resnet-152. The results are shown in Table 8. From the result, one could find that the performance gap between Resnet-50 and Resnet-152 is narrow (about 0.2% in ERR). This is why Resnet-50 is used as a pre-trained model for the feature transfer block.

5.8.3 Coding Block

Two baselines are used to demonstrate the effectiveness of our coding block, namely, **LISTA+MFB+FTB** and **ISTA+MFB+FTB** which replace SC2Net with a simple RNN based ℓ_1-solver (i.e., LISTA) [24] and a traditional sparse coding method (ISTA [73]). Note that, if removing the motion fusion block, then the corresponding method could be regarded as a simplified version of the recent SRNN [5]. The results of these two baselines are summarized in the last two rows in Table 7. From the results, one could observe that SC2Net is superior to the traditional ISTA and the simple RNN based LISTA [24].

5.9 Parameter Analysis

In the proposed model, there are two parameters, i.e., the number of frames to be compressed in the motion fusion block (denote by T), and the sparsity parameter in optimization network (denoted by λ). To investigate their influence on the performance of our method, we conduct the parameter analysis and report the results in Table 9. For the other used datasets, we also have similar observation, namely, $T = 20, \lambda = 0.1$ usually leads to a desirable result in terms of detection quality and robustness.

6 Conclusion

In this paper, we propose a unified deep learning based framework for abnormal event detection. The proposed AnomalyNet consists of three blocks which are designed to achieve three keys of anomaly detection in neural networks. In short, the motion fusion block is designed to keep the temporal and spatial connection between the motion and appearance cues. The feature transfer block is used to extract discriminative features by exploiting the transferability of the neural network from different tasks/domains. The coding block is a novel LSTM to achieve fast sparse coding, which could enjoy fast inference and end-to-end learning. Extensive experiments show the promising performance of our method in image reconstruction and abnormal events detection in surveillance.

References

[1] C. Lu, J. Shi, and J. Jia, "Abnormal event detection at 150 fps in matlab," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 2720–2727.

[2] M. Sabokrou, M. Fathy, M. Hoseini, and R. Klette, "Real-time anomaly detection and localization in crowded scenes," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2015.

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

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

[5] W. Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked rnn framework," in *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

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

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

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

[9] L. Kratz and K. Nishino, "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, 2009, pp. 1446–1453.

[10] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *European conference on computer vision*. Springer, 2006, pp. 428–441.

[11] F. Jiang, J. Yuan, S. A. Tsaftaris, and A. K. Katsaggelos, "Anomalous video event detection using spatiotemporal context," *Computer Vision and Image Understanding*, vol. 115, no. 3, pp. 323–333, 2011.

[12] D. Zhang, D. Gatica-Perez, S. Bengio, and I. McCowan, "Semi-supervised adapted hmms for unusual event detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 611–618.

[13] J. Kim and K. Grauman, "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, 2009, pp. 2921–2928.

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

[15] Y. S. Chong and Y. H. Tay, "Abnormal event detection in videos using spatiotemporal autoencoder," in *International Symposium on Neural Networks*. Springer, 2017, pp. 189–196.

[16] J. T. Zhou, M. Fang, H. Zhang, C. Gong, X. Peng, Z. Cao, and R. S. M. Goh, "Learning with annotation of various degrees," *IEEE Trans. Neural Netw. Learning Syst.*

[17] A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Fast L1-Minimization algorithms and an application in robust face recognition: A review," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2010-13, February 5 2010.

[18] X. Peng, C. Lu, Y. Zhang, and H. Tang, "Connections between nuclear norm and frobenius norm based representation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 1, pp. 218–224, Jan. 2018.

[19] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 629–654, 2008.

[20] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural networks*, vol. 12, no. 1, pp. 145–151, 1999.

[21] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[22] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[23] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, "Learning precise timing with lstm recurrent networks," *Journal of Machine Learning Research*, vol. 3, no. Aug, pp. 115–143, 2002.

[24] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *ICML-10, June 21-24, 2010, Haifa, Israel*, 2010, pp. 399–406.

[25] J. T. Zhou, K. Di, J. Du, X. Peng, H. Yang, S. J. Pan, I. W. Tsang, Y. Liu, Z. Qin, and R. S. M. Goh, "Sc2net: Sparse lstms for sparse coding," in *AAAI 2018, February 2 - 7, 2018, New Orleans, Louisiana, USA.*, 2018.

[26] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[27] T. Xiao, C. Zhang, H. Zha, and F. Wei, "Anomaly detection via local coordinate factorization and spatio-temporal pyramid," in *Asian Conference on Computer Vision*. Springer, 2014, pp. 66–82.

[28] Z. Zhu, J. Wang, and N. Yu, "Anomaly detection via 3d-hof and fast double sparse representation," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 286–290.

[29] H. Mousavi, M. Nabi, H. K. Galoogahi, A. Perina, and V. Murino, "Abnormality detection with improved histogram of oriented tracklets," in *International Conference on Image Analysis and Processing*. Springer, 2015, pp. 722–732.

[30] V. Reddy, C. Sanderson, and B. C. Lovell, "Improved anomaly detection in crowded scenes via cell-based analysis of foreground speed, size and texture," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*. IEEE, 2011, pp. 55–61.

[31] X. Peng, J. Feng, S. Xiao, W. Y. Yau, J. T. Zhou, and S. Yang, "Structured autoencoders for subspace clustering," *IEEE Trans Image Process*, vol. 27, no. 10, pp. 5076–5086, Oct 2018.

[32] Z. Huang, H. Zhu, J. T. Zhou, and X. Peng, "Multiple marginal fisher analysis," *IEEE Trans Industr Electron*, pp. 1–1, 2018.

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

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

[35] M. Sabokrou, M. Fayyaz, M. Fathy, Z. Moayed, and R. Klette, "Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes," *Computer Vision and Image Understanding*, 2018.

[36] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International Conference on Information Processing in Medical Imaging*. Springer, 2017, pp. 146–157.

[37] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection - A new baseline," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 2018, pp. 6536–6545.

[38] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "Ganomaly: Semi-supervised anomaly detection via adversarial training," *arXiv preprint arXiv:1805.06725*, 2018.

[39] M. Sabokrou, M. Pourreza, M. Fayyaz, R. Entezari, M. Fathy, J. Gall, and E. Adeli, "Avid: Adversarial visual irregularity detection," in *Asian Conference on Computer Vision (ACCV), 2018, Perth, Australia, Dec 2-6*, 2018.

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

[41] H. Ren, H. Pan, S. I. Olsen, and T. B. Moeslund, "A comprehensive study of sparse codes on abnormality detection," *CoRR*, vol. abs/1603.04026, 2016. [Online]. Available: http://arxiv.org/abs/1603.04026

[42] J. T. Rolfe and Y. Lecun, "Discriminative recurrent sparse auto-encoders," in *Proceedings of the International Conference on Learning Representations (ICLR) 2013, May 2 - 4, 2013, Scottsdale, USA*, 2013.

[43] P. Sprechmann, A. M. Bronstein, and G. Sapiro, "Learning efficient sparse and low rank models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1821–1833, Sept 2015.

[44] Y. Yang, J. Sun, H. Li, and Z. Xu, "Deep admm-net for compressive sensing MRI," in *NIPS 2016, December 5-10, 2016, Barcelona, Spain*, 2016, pp. 10–18.

[45] S. Diamond, V. Sitzmann, F. Heide, and G. Wetzstein, "Unrolled optimization with deep priors," *arXiv preprint arXiv:1705.08041*, 2017.

[46] Z. Wang, Q. Ling, and T. S. Huang, "Learning deep l0 encoders," in *AAAI 2016, February 12-17, 2016, Phoenix, Arizona, USA.*, 2016, pp. 2194–2200.

[47] W. Zuo, D. Ren, D. Zhang, S. Gu, and L. Zhang, "Learning iteration-wise generalized shrinkage–thresholding operators for blind deconvolution," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1751–1764, 2016.

[48] Z. Wang, S. Chang, Y. Yang, D. Liu, and T. S. Huang, "Studying very low resolution recognition using deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4792–4800.

[49] J. T. Zhou, H. Zhao, X. Peng, M. Fang, Z. Qin, and R. S. M. Goh, "Transfer hashing: From shallow to deep," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 29, no. 12, p. 61916201, 2018.

[50] T. Moreau and J. Bruna, "Understanding neural sparse coding with matrix factorization," *Proceedings of the International Conference on Learning Representations (ICLR)*, April 2017.

[51] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould, "Dynamic image networks for action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3034–3042.

[52] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust real-time unusual event detection using multiple fixed-location monitors," *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 3, pp. 555–560, 2008.

[53] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[54] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *CVPR*, 2014.

[55] J. T. Andrews, "Transfer representation-learning for anomaly detection."

[56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[57] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[58] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.
[59] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via 1 minimization," *Proceedings of the National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003.
[60] A. S. Bandeira, E. Dobriban, D. G. Mixon, and W. F. Sawin, "Certifying the restricted isometry property is hard," *IEEE Transactions on Information Theory*, vol. 59, no. 6, pp. 3448–3450, 2013.
[61] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.
[62] P. Koiran and A. Zouzias, "On the certification of the restricted isometry property," *arXiv preprint arXiv:1103.4984*, 2011.
[63] "Unusual crowd activity dataset of university of minnesota, available from http://mha.cs.umn.edu/movies/crowdactivity-all.avi."
[64] W. Luo, W. Liu, and S. Gao, "Remembering history with convolutional lstm for anomaly detection," in *Multimedia and Expo (ICME), 2017 IEEE International Conference on*. IEEE, 2017, pp. 439–444.
[65] S. Smeureanu, R. T. Ionescu, M. Popescu, and B. Alexe, "Deep appearance features for abnormal behavior detection in video," in *Image Analysis and Processing - ICIAP 2017*, S. Battiato, G. Gallo, R. Schettini, and F. Stanco, Eds. Cham: Springer International Publishing, 2017, pp. 779–789.
[66] R. V. H. M. Colque, C. Caetano, M. T. L. de Andrade, and W. R. Schwartz, "Histograms of optical flow orientation and magnitude and entropy to detect anomalous events in videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 3, pp. 673–682, 2017.
[67] M. Ravanbakhsh, M. Nabi, E. Sangineto, L. Marcenaro, C. Regazzoni, and N. Sebe, "Abnormal event detection in videos using generative adversarial nets," *arXiv preprint arXiv:1708.09644*, 2017.
[68] D. Xu, Y. Yan, E. Ricci, and N. Sebe, "Detecting anomalous events in videos by learning deep representations of appearance and motion," *Computer Vision and Image Understanding*, vol. 156, pp. 117–127, 2017.
[69] M. Sabokrou, M. Khalooei, M. Fathy, and E. Adeli, "Adversarially learned one-class classifier for novelty detection," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 2018, pp. 3379–3388.
[70] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 935–942.
[71] W. Li, V. Mahadevan, and N. Vasconcelos, "Anomaly detection and localization in crowded scenes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 1, pp. 18–32, 2014.
[72] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proceedings*. Presses universitaires de Louvain, 2015, p. 89.
[73] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research*, vol. 11, no. Jan, pp. 19–60, 2010.