SVD-GAN for Real-Time Unsupervised Video Anomaly Detection

Dinesh Jackson Samuel

Fabio Cuzzolin
https://cms.brookes.ac.uk/staff/FabioCuzzolin/

Faculty of Technology, Design and Environment
Visual Artificial Intelligence Laboratory
Oxford Brookes University
Oxford, UK

Abstract

Real-time unsupervised anomaly detection from videos is challenging due to the uncertainty in occurrence and definition of abnormal events. To overcome this ambiguity, an unsupervised adversarial learning model is proposed to detect such unusual events. The proposed end-to-end system is based on a Generative Adversarial Network (GAN) architecture with spatiotemporal feature learning and a new Singular Value Decomposition (SVD) loss function for robust reconstruction and video anomaly detection. The loss employs efficient low-rank approximations of the matrices involved to drive the convergence of the model. During training, the model strives to learn the relevant normal data distribution. Anomalies are then detected as frames whose reconstruction error, based on such distribution, shows a significant deviation. The model is efficient and lightweight due to our adoption of depth-wise separable convolution. The complete system is validated upon several benchmark datasets and proven to be robust for complex video anomaly detection, in terms of both AUC and Equal Error Rate (EER).

1 Introduction

Anomalies can be defined as unusual events which deviate from ‘normal’ behaviour. Anomaly detection is relevant to various applications, including intrusion detection in time series, surveillance, action detection and healthcare monitoring. Manually detecting anomalies from surveillance videos is a painstaking job, and a demanding one in terms of human resources. With the increase of surveillance cameras the need for an automated system for anomaly detection from videos has thus gained much recognition. These systems play a crucial role in security control, crime detection, accidents and traffic monitoring, where the data available from various sources is simply too much for manual analysis.

The occurrence of an anomalous event is unexpected and rare in practice, making it difficult to categorise such diverse events. For instance, people running on the beach is considered normal behaviour, whereas running in a shopping mall is considered anomalous. Therefore, automated system will not have any prior knowledge about the nature of past or future anomalies. Representational learning methods, such as sparse-coding, achieve good performance in anomaly detection [20, 46]. Anomaly detection in videos, in particular, can be posed in either the supervised or the unsupervised learning setting. In supervised
anomaly detection, the system learns from example videos labelled as either anomalous or non-anomalous [17, 37, 45]. In the unsupervised approach, instead, the system assumes any rare or abnormal occurrence, deviating from the learned normal sample parameters, to be anomalous [38]. The model can thus be trained to detect anomalies using huge volumes of unlabeled, ‘normal’ data. In surveillance, variations due to, for instance, changes in scale and viewpoint, introduce additional ambiguity.

A number of studies have focused on the use of Convolutional Neural Networks (CNNs) for (supervised) anomaly detection in industrial products [43], for instance inspecting cement surfaces [3] and cracks [5]. In the supervised setting, however, the issue arises of an often uneven balance between normal and abnormal data. The use of data augmentation has been proposed to mitigate this challenge – nevertheless, this setting still has serious limitation in addressing real life problems [30, 47]. In opposition, Ravanbakhsh et al. have more recently proposed to employ adversarial learning to localise anomalous activities in an unsupervised setting [30]. Generative Adversarial Networks (GANs), in particular, as they have the capability to model high-dimensional image data, have become the state of the art in anomaly detection in recent times. Many GAN-centric architectures, such as AnoGAN [35] and GANomaly [2], have been put forward to this purpose. The objective of GAN, however, which encourages the generated samples to look like real data, is not directly aligned with the goal of performing anomaly detection. Consequently, in much recent work in anomaly detection adversarial training has been modified to improve both training and inference for this specific goal [2, 35]. Overall, both CNN-based and GAN-based architectures are inefficient to run on edge devices such as robots, smart surveillance cameras, autonomous driving cars or microcomputers. In addition, most of the networks used in unsupervised GAN architectures are shallow and are designed to learn only spatial features, ignoring the crucial temporal component of videos. As a result, such networks are tapered to low-dimensional data and prone to overfitting because of the large number of parameters.

Contributions. In this paper, inspired by the GANomaly [2] architecture but significantly departing from it, we propose a light-weight, efficient anomaly detection architecture with a reduced number of parameters, aimed at addressing the convergence and overfitting problems with GAN training and at achieving real-time performance. Our SVD-GAN has an encoder-decoder architecture as the backbone, and is capable of learning in an unsupervised way both spatial and temporal features from real-time videos, thanks to a stacked convolutional Long-Short Term Memory (LSTM) network structure. Our main contributions are:

• An original Singular Value Decomposition (SVD) loss function which backpropagates only a small number of ‘dominant’ patterns from the input and generated video frames as loss value, thus improving the reconstruction ability of the generator.

• An original generator structure making use of depth-wise convolution layers of our own design, which leads to increased efficiency due to a reduction in the number of model parameters of 15.9% without compromising on feature extraction, yielding a model that is both lightweight and more stable.

Whereas using the proposed SVD loss in GANs drastically improves their performance (as it minimises the KL divergence between real and generated data distributions), the new loss can be widely employed in other deep learning models for better representation learning, whenever few training samples are available (e.g., in few shot learning). To avoid the vanishing gradients issue, network parameters are updated via distinct sub gradients [12] using residual connections, in an original architecture for spatiotemporal feature extraction from
videos. Our system can detect complex anomalous events occurring for a very short time, and outperforms the prior art on the large-scale CHUK Avenue and ShanghaiTech datasets.

2 Background

Deep learning for anomaly detection. Deep learning-based methods have achieved success in detecting abnormal events from videos, outperforming the former state of the art in challenging environments [1, 3, 9, 18, 31, 36]. Deep neural networks with hierarchical feature representation learning are simply more powerful than the handcrafted feature extraction techniques used in traditional architectures. Deep generative models, in particular, have recently come to the fore with their ability to encode complex transformations. Liu et al. proposed the use of GANs to detect anomalies by minimising the difference between predicted future video frames and ground truth frames [19]. During training the normal data distribution is estimated using the training video frames $X_{\text{train}} = \{X_i\}$ by learning a parametric representation $f_\theta : X_{\text{train}} \to \mathbb{R}^m$ which minimises the model reconstruction loss. At test time, an anomaly score $A(X_j)$ is computed for each test frame $X_j \in X_{\text{test}}$ as the deviation from the learnt optimal representation $f_{\theta^*}$:

$$A(X_j) = \|f_{\theta^*}(X_j) - X_j\|^2.$$  
Finally, anomalies are detected by applying a threshold $T$, $A(X_j) > T$, to the anomaly score.

Generative Adversarial Networks consist of two different networks, a generator and a discriminator, both trained with unlabeled data [10, 29, 32]. The generator $G$ aims to capture the data distribution and generate realistic video frames, by building a data distribution for the input data $X$ via a mapping from a prior latent space noise distribution $z$. The objective of the discriminator $D$, instead, is to find the probability of the sample being outputted by the generator. Generator and discriminator compete against each other by playing a zero-sum min-max game:

$$V(D, G) = E_{X \sim p_{\text{data}}}(X) \log D(X) + E_{Z \sim p_z}(Z) \log(1 - D(G(Z))).$$

In recent years, various anomaly detection GAN architectures have been proposed. One such architecture is the extended conditional GAN proposed by Mizra et al. [25]. This model conditions either the generator $G$ or the discriminator $D$ using some additional information $Y$. The condition $Y$ can be formulated from multimodal input data or class labels available as auxiliary information. Vu et al. [41] have proposed a robust anomaly detection system for videos which uses conditional GANs to detect video anomalies accurately at various levels of representation using a layer-wise approach. The extraction of optical flow data poses challenges in uncertain environments characterised by untextured regions, illumination changes, occlusions and fast motions. The Bidirectional GAN (BiGAN) architecture proposed by Donahue et al. [8] consists of an encoder $E(X, E(X))$ which learns the inverse mapping of the generator, $E = G - 1$. During training, the model learns how to map the latent space to the image data and vice-versa, mitigating the statistical complexity of the process and producing better results on the MNIST benchmark dataset [8].

In 2019, a new mapping scheme called ‘fast-AnoGAN’ was proposed by Schlegl et al. [34], which can perform both fast detection of anomalies at image level and pixel-level localisation. In 2018, Zenati et al. proposed an Efficient GAN-Based Anomaly Detection (EGBAD) system which leverages the BiGAN architecture [44]. Ackay et al., instead, hypothesized the learning of image space and latent space vectors jointly. The architecture employs an adversarial autoencoder with an encoder-decoder-encoder pipeline for capturing the data distribution of image and latent space vectors. This architecture, however, is limited in the way it handles spatial-temporal learning, and produces unstable reconstructions for real-time videos [9]. Vu et al. proposed a robust anomaly detection system for videos which uses representational learning from both intensity and motion information via condi-
tional GANs [41]. Nguyen and Meunier [27] proposed a deep CNN that addresses anomaly detection by learning a correspondence between common object appearances (e.g., pedestrian, background, tree, etc.) and their associated motions. A combination of future frame prediction and reconstruction for anomaly detection was proposed by Tang et al. using two U-Net blocks in the generator [1], where the first block tries to predict frames while the other reconstructs the frames generated by the former block.

**Convolutional LSTMs.** Jefferson et al. proposed a Conv-LSTM network in an encoder-decoder model for the prediction of future frames and anomaly detection by reconstruction [24]. The same architecture was proved to be promising for video anomaly detection in [22]. Input video frames are there passed to the convolutional LSTM for feature extraction and then reconstructed using deconvolution. Luo et al. proposed a Temporally-coherent Sparse Coding (TSC) approach in which similar neighbouring frames are mapped via stacked Recurrent Neural Networks (RNNs) to a reconstruction coefficient [21]. LSTM autoencoders are indeed suited to the extraction of spatial-temporal information. Shi et al. [36] and Patraucean et al. [28] both used stacked convolutional LSTMs in an autoencoder architecture for feature extraction in video sequence data. Conv-LSTMs can capture both spatial representations and spatio-temporal information from a sequence of video frames, and display a marked ability to predict future frames.

In Conv-LSTMs the amount of information from the previous time step received by the hidden state is partly determined by the size of convolutional filter in the hidden-to-hidden connection. To capture faster motions large transitional kernels are used, while for slower motions small kernels can do [41]. GAN performance, however, degrades drastically as the number of parameters increases, leading GANs to often fail on high dimensional data. To overcome this, in our model *depth-wise separable convolutions* are used within the LSTM, reducing the size of the model and the chance of overfitting during GAN training. Depth-wise separable convolution is followed by pointwise convolution to deal with the spatial and depth dimensions of the video frames, thus splitting a $3 \times 3$ kernel into $3 \times 1$ and $1 \times 3$ kernels. Each input frame is processed using three separate filters for the R, G and B channels.

### 3 Architecture and methodology

![Proposed spatiotemporal GAN architecture for video anomaly detection.](image)

Figure 1: Proposed spatiotemporal GAN architecture for video anomaly detection.
Given a number of unlabeled sample video frames \( \mathbf{X}_{\text{train}} = (X_1, \ldots, X_n) \) at training time, our unsupervised anomaly detection approach is designed to learn suitable spatiotemporal features from \( \mathbf{X}_{\text{train}} \) that, at test time, it employs to detect whether a new video frame \( X \) is anomalous via an anomaly score \( A(X) \). The models learn from \( \mathbf{X}_{\text{train}} \) the probability distribution of the ‘normal’ frames while minimising the anomaly score of the training samples.

### 3.1 Architecture

Our proposed architecture (Figure 1) is based on the generative adversarial network principle and uses an encoder-decoder-encoder [9] pipeline as the Generator, which learns feature representations directly from the input samples, and an encoder-based Discriminator which aims to discern real from fake images.

At each time step, a batch of video frames of fixed duration is passed as input to both the generator \( G \) and the discriminator \( D \). Each input image \( X \) is passed to the encoder \( E \) in the generator \( G \), which maps it to a latent space \( Z = E(X) \). The decoder then transposes these latent space vectors back to the image data space, implementing a mapping \( \bar{X} = G(X) \).

The first contribution of this paper is an original Generator architecture, shown in Figure 2, where it can be seen how the generator learns the temporal dependencies within a video sequence using temporal blocks of frames. The architecture is fully described in Sec. 3.4.

![Figure 2: Pipeline of our spatiotemporal GAN generator](image)

At training time, the network learns the joint posterior distribution of the data \( G(Z, X) \), and each input sample \( X \) is encoded using its latent representation. Training is performed using the inverse mapping from image data to latent space proposed by Lipton et al [18]. The generated image \( \bar{X} = G(X) \) and the input image \( X \) are then passed to the discriminator, which discriminates real from generated frames using a sigmoid activation function and backpropagates the loss to the generator itself.

### 3.2 Singular Value Decomposition (SVD) loss

GANs are known to have convergence issues with high-dimensional input data (e.g., video frames). Vanishing gradient problems tend to occur in the generator during training.

**Minimising the KL divergence.** GAN training, however, can be improved by minimising the Kullback-Leibler (KL) divergence between the ‘real’ data distribution \( P_{\text{data}}(X) \) and the ‘generated’ distribution \( P_{\text{gen}}(X) = P(G(X)) \), i.e., the probability distribution of the generated images [9]:

\[
KL(P_{\text{data}} \parallel P_{\text{gen}}) = \int_X P_{\text{data}}(X) \log \frac{P_{\text{data}}(X)}{P_{\text{gen}}(X)} dX = E_{X \sim P_{\text{data}}(X)} [\log P(X) - \log P(G(X))].
\]
Making the generated distribution as close as possible to the real one is indeed the main objective of GAN. Various measures of dissimilarity as possible, including total variation and the Wasserstein distance [3].

In this paper, rather than posing as an explicit objective the minimisation of the KL divergence of the distributions of real and generated data, we aim to achieve the above goal through an original Singular Value Decomposition (SVD) loss function designed to back-propagate the difference between low-rank approximations of the input $X$ and generated $G(X)$ images, defined by their principal components.

**Properties of SVD decomposition.** SVD is a matrix decomposition technique that, when applied to images, is able to encode the maximum fraction of signal energy using only a few coefficients [4, 26, 31]. An input image matrix $X$ (e.g., a video frame) with $m$ rows and $n$ columns factorises as

$$X = USV^T,$$

where $U$ is an orthogonal $m \times m$ matrix, $V$ is orthogonal and $n \times n$, and $S$ is a matrix of the same size as $X$, with on the diagonal the singular values $\sigma_i$ of $X$ (see Fig. 3). The columns $u_i$ of $U$ are the eigenvectors of $XX^T$, while the columns $v_i$ of $V$ are the eigenvectors of $X^TX$.

A fundamental property of SVD is that the matrix

$$\hat{X}_r = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

(with the eigenvalues in order of magnitude) is the optimal solution to the problem $\min \|X - X'\|_F$ (where $F$ denotes the Frobenius norm) subject to rank$(X') \leq r$, i.e., is the optimal rank-$r$ approximation of $X$.

![Figure 3: Graphical representation of the SVD decomposition of a matrix (frame) $X$.](image)

**SVD loss.** Our proposal is to minimise the empirical expectation of the $L_2$ norm of the difference between the low-rank SVD approximations of the input image $X$ and of the generated image $G(X)$, which we term SVD loss:

$$SVD_{loss} = \mathbb{E}_{X \sim p(X)} \|\hat{X}_r - \hat{G(X)}_r\|_2,$$

where $\hat{X}_r$ and $\hat{G(X)}_r$ are the rank-$r$ approximations of $X$ and $G(X)$, respectively.

Our conjecture, which we empirically validate in this paper, is that minimising the SVD loss is indeed correlated with minimising the KL divergence between real and generated data, and should thus have positive effects on the convergence of our model.

### 3.3 Overall loss

In our model, more stable GAN training is achieved by using as overall loss:

$$SVD_{loss} + ADV_{loss} + CONT_{loss} + ENCOD_{loss}.$$  

The adversarial loss $ADV_{loss}$ is the Jensen-Shannon divergence of the output of the discriminator $D$ for the input image $X$ and the corresponding generated image $G(X)$ [32], namely:

$$ADV_{loss} = \mathbb{E}_{X \sim p(X)} |D(X) - D(G(X))|.$$  

A contextual loss based on the $L_1$ norm is used for
penalising the distance between the input image \(X\) and the reconstructed image \(G(X)\) \([13]\): \(\text{CONT}_{\text{loss}} = \mathbb{E}_{X \sim \mathbb{P}(X)} |X - G(X)|_1\). Our SVD loss can thus be seen as an efficient form of contextual loss, based on the \(L_2\) norm. Compared to the \(L_1\) norm, \(L_2\) better penalises outliers, as deviations are magnified by taking the square. Finally, the encoder loss minimises the distance between the bottleneck features of the input \(Z = E(X)\) and the encoded features of the generated image \(\tilde{Z} = E(G(X))\), namely: \(\text{ENCOD}_{\text{loss}} = \mathbb{E}_{X \sim \mathbb{P}(X)} |Z(X) - \tilde{Z}(X)|_2\).

### 3.4 Generator structure and training protocol

We trained our spatiotemporal GAN model on an 8-GPU machine with Quadro RTX 6000 cards having 24 GB VRAM each. Input frames were resized to 128 × 128 pixels and passed to the SVD-GAN architecture. The proposed architecture uses \(\text{tanh}\) activations in the generator and \(\text{LeakyRelu}\) ones in the discriminator. Batch normalisation with \(\text{tanh}\) at the end of each layer helps scaling and adjusting the input features to the interval \([-1, 1]\). The generator uses an Adam optimiser with first-order derivatives. The discriminator uses RMSProp with a 0.00005 learning rate for weight optimisation.

In our Generator, each batch of \(n\) rescaled input frames goes through two layers of depthwise convolution and 4 layers of convolutional LSTM for spatio-temporal feature learning, as shown in Figure 2. Frames are convolved with a kernel of size 5 × 5 and stride 2 to produce a feature map of size \(n \times 64 \times 64 \times 128\). Subsequently, a small kernel of size 3 × 3 is applied to the respective feature maps to capture spatiotemporal features using a block of convolutional LSTMs. Input frames are encoded to a latent space \(Z\) of size \(n \times 16 \times 16 \times 128\) to be then passed to the decoder for reconstruction. The decoder uses convolutional 2D transpose layers and batch normalisation to decode the bottleneck features back to the image space \(\tilde{X}\). The reconstructed data is remapped to the latent space \(\tilde{Z}\) for a consistent comparison between \(Z\) and \(\tilde{Z}\). Finally, the generated image \(\tilde{X}\) and the input image \(X\) are given as inputs to the discriminator which has the same encoder architecture as the generator, with an additional sigmoid activation for discrimination. Losses are back propagated to the generator for an accurate reconstruction of the input image \(X\).

### 3.5 Anomaly detection

During testing, anomalies are detected by thresholding an anomaly score \(A(X)\), the square \(L_2\) distance between input and reconstructed images, rescaled to the interval \([0, 1]\):

\[
A(X) = \frac{1}{p} \sum_{i=1}^{p} (\tilde{X}_j(i) - X_j(i))^2, \quad \hat{A}(X) = \frac{A(X) - A_{\min}}{A_{\max}},
\]

where \(X_j\) is the input test video frame, \(\tilde{X}_j\) is the reconstructed video frame, \(p\) the number of pixels in a frame, and \(A_{\min} / A_{\max}\) are its minimum / maximum over the test sequence.

Note that we do not commit to any specific threshold, but assess the performance of a model over the whole range of thresholds by measuring the Area Under the ROC Curve (AUC), after plotting the model’s Receiver Operating Characteristics (ROC) curve (see Section 4.1 - Metrics).

### 4 Experiments

#### 4.1 Datasets and evaluation metrics

We validate our approach over several benchmark datasets portraying complex anomalous events in various scenarios involving multiple scenes captured from different angles. All
datasets comprise ‘normal’ video frames for training and a combination of anomalous and non-anomalous frames for testing. Their features are summarised in Table 1.

The CHUK Avenue dataset contains 16 normal videos for training and 21 videos for testing, for a total of 30,652 frames [20]. Test videos include anomalies like the throwing of objects, walking in the wrong direction, running, and loitering. The UCSD anomaly detection dataset contains surveillance videos of pedestrian walkways [23]. Anomalies include presence of skaters, bikers, small carts and people walking sideways in walkways. The dataset is divided into two parts: Ped1 and Ped2. Ped1 contains 34 normal video samples for training with some perspective distortion and 36 videos samples for testing. Ped2 portrays pedestrians walking parallelly to the camera plane, with 16 videos for training and 12 for testing. The complex ShanghaiTech Campus dataset is specifically used to validate the robustness of the model [19]. The corpus contains video sources from 13 different scenes, under various lighting conditions and camera angles. It has 330 video samples for training and 107 for testing, with a total of 130 abnormal events. Anomalies include complex unusual human behaviors, presence of unusual objects and movements in the wrong direction.

<table>
<thead>
<tr>
<th>Benchmark Datasets</th>
<th>Anomalous Events</th>
<th>Source Frames</th>
<th>Normal Frames</th>
<th>Anomalous Frames</th>
<th>Training Frames</th>
<th>Testing Frames</th>
</tr>
</thead>
<tbody>
<tr>
<td>UCSD Ped1</td>
<td>40</td>
<td>1</td>
<td>9995</td>
<td>4005</td>
<td>6800</td>
<td>7200</td>
</tr>
<tr>
<td>UCSD Ped2</td>
<td>12</td>
<td>1</td>
<td>2924</td>
<td>1636</td>
<td>2550</td>
<td>2010</td>
</tr>
<tr>
<td>CHUK Avenue</td>
<td>47</td>
<td>1</td>
<td>26832</td>
<td>3820</td>
<td>15328</td>
<td>15324</td>
</tr>
<tr>
<td>ShanghaiTech</td>
<td>130</td>
<td>13</td>
<td>300308</td>
<td>17090</td>
<td>274515</td>
<td>42883</td>
</tr>
</tbody>
</table>

Table 1: Characteristics of the datasets used in this work.

Metrics. The model’s frame-level performance is analysed using the Area Under the ROC Curve (AUC), after plotting the Receiver Operating Characteristics (ROC) curve. The latter plots the True Positive Rate (TPR) vs the False Positive Rate (FPR) as a function of the detection threshold in the range [0,1], thus summarising the trade-off between TPR and FPR for a predictive model using different probability thresholds.

AUC measures the two-dimensional area under the entire ROC curve from (0,0) to (1,1), providing an aggregate measure of performance across all possible detection thresholds which amounts to a sort of probability distribution over the range of thresholds. The AUC thus represents the degree of separability the model can enforce between anomalous and non-anomalous frames. The higher the AUC value, the better the performance.

The EER (Equal Error Rate), the point on the ROC curve where false positive and false negative rates coincide, is also reported (see Table 2).

4.2 Comparison with state of the art

The performance of our SVD-GAN on all datasets is compared with that state-of-the-art unsupervised anomaly detection systems in Table 2. The table also compares the performance of our proposed architecture with and without SVD loss. The architecture with SVD loss achieves state-of-the-art results, by a very large margin, on both the Avenue and the ShanghaiTech datasets, with an AUC of 89.82% and EER of 21.55% on CHUK Avenue and an AUC of 78.42% and EER of 25.16% on ShanghaiTech. The latter, in particular, compared to all other datasets is a truly large scale benchmark with 130 very complex anomalous events.

Detailed information on the number of frames and anomalous events for each dataset is provided in the supplementary material. On UCSD our model’s AUC is low compared to the state of the art. This is likely due to the relatively short duration (ca 200 frames) of
<table>
<thead>
<tr>
<th>Unsupervised Methods</th>
<th>UCSD Ped1</th>
<th>UCSD Ped2</th>
<th>Avenue</th>
<th>ShanghaiTech</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AUC</td>
<td>EER</td>
<td>AUC</td>
<td>EER</td>
</tr>
<tr>
<td>MPPC+SFA [2010] [23]</td>
<td>66.8</td>
<td>–</td>
<td>61.3</td>
<td>–</td>
</tr>
<tr>
<td>Conv-AE [2016] [11]</td>
<td>81.1</td>
<td>27.9</td>
<td>90.0</td>
<td>21.7</td>
</tr>
<tr>
<td>ConvLSTM-AE [2017] [22]</td>
<td>75.5</td>
<td>–</td>
<td>88.1</td>
<td>–</td>
</tr>
<tr>
<td>sRNN [2017] [21]</td>
<td>–</td>
<td>–</td>
<td>92.21</td>
<td>–</td>
</tr>
<tr>
<td>STAN [2018] [21]</td>
<td>82.1</td>
<td>–</td>
<td>96.5</td>
<td>–</td>
</tr>
<tr>
<td>Flownet+UNet [2017] [19]</td>
<td>83.1</td>
<td>–</td>
<td>95.4</td>
<td>–</td>
</tr>
<tr>
<td>MLAD [2019] [41]</td>
<td>82.34</td>
<td><strong>23.50</strong></td>
<td><strong>99.21</strong></td>
<td><strong>2.49</strong></td>
</tr>
<tr>
<td>ITAE+NFs [2020] [16]</td>
<td>–</td>
<td>–</td>
<td>97.3</td>
<td>–</td>
</tr>
<tr>
<td>ROADMAP [2021] [42]</td>
<td><strong>83.4</strong></td>
<td>–</td>
<td>96.3</td>
<td>–</td>
</tr>
<tr>
<td>Ours without SVD loss</td>
<td>70.53</td>
<td>31.20</td>
<td>74.6</td>
<td>25.32</td>
</tr>
<tr>
<td>Ours with SVD loss</td>
<td>73.26</td>
<td>28.75</td>
<td>76.98</td>
<td>23.46</td>
</tr>
</tbody>
</table>

Table 2: Performance comparison in terms of AUC and EER among state-of-art unsupervised anomaly detection architectures, including ours.

the available training sequences. In this situation the model tends to reconstruct well the anomalous frames too and falls short when detecting certain anomalies. E.g., the AUC is comparatively low for videos portraying skaters or cyclists in pedestrian pathways – these anomalies look closer to normal from the angle (top view) from which the video is captured. This behaviour is to be expected, for LSTM-based models need sufficient sequential data to be excited and perform well. A graph plotting the regularity score vs the time stamp for a test video from Shanghai Tech and Avenue datasets are shown in the supplementary material.

The variance of the processing time of frames from the mean is 4ms, and the worst-case performance (per frame) is 27ms. The overall Floating Point Operations (FLOPs) of our SVD-GAN is also relatively low, at $2.214042 \times 10^6$ FLOPs.

### 4.3 Overfitting and accurate reconstruction

The proposed tackles two key issues preventing a GAN-based architecture from reliably reconstructing video frames. Overfitting is addressed by reducing the number of parameters via depth-wise separable convolution. Compared to 2D convolution, depth-wise convolution reduces the encoder’s parameters by 4.3% and the decoder’s by 3%, with a total parameter reduction of 15.9%, without any painful trade-off for feature learning, efficiently reducing the parameters in the spatial feature extraction layers (i.e., Layers 1 and 2) by a factor of $n$. This makes the model sufficiently lightweight and efficient for real time anomaly detection.

Secondly, the use of the original $SVD_{loss}$ in the generator allows us to minimise the distance between low-rank optimal approximations of input and generated images, aiding network convergence. As a result our SVD-GAN model starts converging as early as the 167th epoch (with LeakyReLU as normaliser), gradually attaining a stable reconstruction characterised by low error values for ‘normal’ frames.

Figure 4 illustrates the anomaly detection graph, with highlighted the frames which are false positives and true positives, for a sequence of the ShanghaiTech dataset. The robustness of the proposed system’s reconstruction abilities is visually illustrated in Fig. 5. More examples of reconstructed video frames are shown in Fig. S2 of the supplementary material.
Figure 4: Plot of the anomaly detection graph, with highlighted frames which are false positives and true positives, for a test sequence of the ShanghaiTech dataset.

Figure 5: Examples of frames reconstructed by SVD-GAN, compared to the original inputs.

5 Conclusions

The proposed lightweight SVD-GAN architecture has a clear edge over state-of-art unsupervised anomaly detection methods while using fewer parameters, thanks to the use of temporal blocks for better spatiotemporal feature learning and an original SVD loss for more robust GAN learning. Our experiments show that our system widely outperforms prior art on both the Avenue and ShanghaiTech datasets and can leverage large-scale datasets. In the future, the accuracy of the system can be further improved by using a memory module or a state-of-the-art 3D feature extractor.

Acknowledgement

This project has received funding from the European Union’s Horizon 2020 research and innovation programme, under grant agreement No. 779813 (SARAS).
References


