Simple and Effective Prevention of Mode Collapse in Deep One-Class Classification

Penny Chong*, Lukas Ruff†, Marius Kloft§, and Alexander Binder∗

*ISTD Pillar, Singapore University of Technology and Design, Singapore
†Machine Learning Group, Technische Universität Berlin, Berlin, Germany
§Department of Computer Science, Technische Universität Kaiserslautern, Kaiserslautern, Germany
Email: penny_chong@mymail.sutd.edu.sg

Abstract—Anomaly detection algorithms find extensive use in various fields. This area of research has recently made great advances thanks to deep learning. A recent method, the deep Support Vector Data Description (deep SVDD), which is inspired by the classic kernel-based Support Vector Data Description (SVDD), is capable of simultaneously learning a feature representation of the data and a data-enclosing hypersphere. The method has shown promising results in both unsupervised and semi-supervised settings. However, deep SVDD suffers from hypersphere collapse—also known as mode collapse—, if the architecture of the model does not comply with certain architectural constraints, e.g. the removal of bias terms. These constraints limit the adaptability of the model and in some cases, may affect the model performance due to learning sub-optimal features. In this work, we consider two regularizers to prevent hypersphere collapse in deep SVDD. The first regularizer is based on injecting random noise via the standard cross-entropy loss. The second regularizer penalizes the minibatch variance when it becomes too small. Moreover, we introduce an adaptive weighting scheme to control the amount of penalization between the SVDD loss and the respective regularizer. Our proposed regularized variants of deep SVDD show encouraging results and outperform a prominent state-of-the-art method on a setup where the anomalies have no apparent geometrical structure.

Index Terms—deep SVDD, mode collapse, anomaly detection

I. INTRODUCTION

Anomaly detection [1] has long been an active area of research [2], [3]. It refers to the detection of unusual events or instances in the data that do not conform to the normal behavior [1]. Anomaly detection algorithms are used extensively in a wide range of applications such as network intrusion detection systems [4], fraud detection systems [5], and the detection of cyber attacks in Cyber-Physical systems (CPS) [6]. In principal, one can frame anomaly detection as a binary supervised classification problem, but such an approach has several disadvantages. An algorithm which learns to classify directly the normal versus anomalous samples has no guarantee to generalize well to new types of anomalies or threats. It is also challenging to collect all possible types of anomalies/threats due to their diversity, lack of labels, or the cost to obtain. Addressing this, Steinwart et al. (2005) [7] assume a uniform distribution for anomalous points, but their approach suffers from the curse of dimensionality. To detect new anomalies, many of these systems adopt an unsupervised learning approach as foundation for their detection algorithm. Some approaches involve detection based on reconstruction errors [8], [9] and some are based on principles derived from classification approaches [10]–[12]. Recently [13] proposed an unsupervised one-class classification approach—deep SVDD, which jointly learns the feature representation of the data and a data-enclosing hypersphere. The unsupervised deep SVDD model has been generalized to a semi-supervised deep SVDD model [14]. However, both models suffer from hypersphere collapse if the architecture of the model does not comply with a set of architectural constraints. The imposed set of constraints include the use of only unbounded activation functions and the removal of bias terms in the network architecture [13]. Imposing these restrictions may cause the network to learn sub-optimal features only. A conceptually related problem to hypersphere collapse is the mode collapse problem observed in GANs [15], [16]. In GANs, the generated samples tend to collapse to a limited number of modes without proper regularization. In the case of deep SVDD, the learned features collapse to a single point if the architectural constraints are not enforced. We use the two terms—mode and hypersphere collapse—interchangeably in this work. We provide related work on the mode collapse problem in GANs in Section II.

In this paper, we address the hypersphere collapse problem in deep SVDD as follows. We propose two regularizers based on random noise injections and minibatch variance, respectively. The resulting regularized variants of deep SVDD form a simple, yet effective regularization against hypersphere collapse in deep SVDD, thereby allowing the removal of architectural constraint on the bias term and activation. The following summarizes our contributions in this work.

1) We propose a regularization technique against hypersphere collapse based on noise injection via the standard cross-entropy loss.

2) We propose a regularization technique against hypersphere collapse that penalizes the minibatch variance if it falls below some annealed threshold.

3) We propose an adaptive weighting scheme to control the balance between the deep SVDD loss and the regularizer.
4) We show that our regularized deep SVDD variants generally perform better than the standard deep SVDD and other methods for both unsupervised and semi-supervised settings.

II. RELATED WORK

Anomaly detection. Approaches in this area can be broadly divided into reconstructive and classification-based methods. For reconstructive methods, usually with autoencoders or GANs, anomalies can be detected from analysing the reconstruction errors since the model only learns to reconstruct normal samples during training. In [8], the authors proposed the robust deep autoencoders (RDA) method which is a combination of autoencoders and robust principle component analysis (RPCA) [17] for anomaly detection. A more advanced approach [18] handles the dataset contamination problem by adapting an adversarial autoencoder architecture to impose a prior distribution on the latent representation by placing anomalies into low likelihood-regions. On the other hand, the authors in [9] are the first to employ a GAN approach for anomaly detection. They introduce the unsupervised AnoGAN model to learn a manifold of the normal samples with generative and discriminative components. Other works that adopt GANs for anomaly detection are [19]–[21].

Another class of anomaly detection algorithms are classification-based approaches. The isolation forest (IF) [12] and the one-class support vector machine (OC-SVM) [10] are unsupervised methods. The kernel-based SVDD [11] which learns a data-enclosing hypersphere of a minimum volume in feature space is inspired by the OC-SVM. The deep SVDD model [13] improves the kernel-based SVDD model by simultaneously learning the feature representation of the unlabeled data and the minimum volume data-enclosing hypersphere. The unsupervised deep SVDD method has also been generalized to the semi-supervised setting [14]. In [22], the authors learn to solve the auxiliary task of classifying different geometric transformations of the normal data. The anomalies are then identified based on their softmax activation statistics of the transformed images. Their approach achieves a new state-of-the-art performance on the CIFAR10 dataset, significantly outperforming some prominent reconstruction and classification-based approaches, including the standard deep SVDD model [13].

Mode collapse in GANs. The hypersphere collapse problem in deep SVDD is conceptually similar to the mode collapse problem in GANs. Deep generative models such as GANs, are known to be difficult to train, unstable, and prone to mode collapse [15], [16]. Without proper regularization, the generator often collapses and thus, generates only samples of a few modes or a single mode of class. In [23], the authors argue that the training issues in GANs are due to the lack of control over the discriminator, and the naturally disjoint manifolds of the actual and generated data. They add a mode regularizer to the generator’s optimization target in an attempt to encourage the generated samples to move towards a nearby mode of the data generating distribution. On the other hand, [16] introduces a set of training methods such as the minibatch discrimination method to improve the training of GANs. Subsequently, [24] proposes an additional reconstructor network that learns to map from the input data distribution to Gaussian random noise. By training the generator and reconstructor networks simultaneously via an implicit variational principle in VEEGAN, the reconstructor network learns the mapping from the data distribution to Gaussian and an approximate inverse of the generator, thereby preventing mode collapse. The work [25] proposes a latent-space constraint loss to prevent mode collapse in Boundary Equilibrium Generative Adversarial Network (BEGAN) [26].

In this work, we compare our proposed regularized variants of deep SVDD with the state-of-the-art method [22] as well as other classification-based anomaly detection methods.

III. PRELIMINARIES

Deep SVDD [13], [14] is an extension of the kernel-based SVDD method [11]. The kernel-based SVDD learns a data-enclosing hypersphere of minimum volume in feature space from the unlabeled training data. However, the kernel method requires manual feature engineering and scales poorly due to the computation/manipulation of the kernel matrix. Deep SVDD overcomes these limitations by jointly learning a suitable feature representation of the data and mapping them to a minimum volume hypersphere [13], [14]. Nonetheless, under certain conditions, deep SVDD is not fully robust against mode collapse, which will be elaborated in Section VI.

A. Unsupervised Deep SVDD

For some input space $\mathcal{X} \subseteq \mathbb{R}^d$ and output space $\mathcal{F} \subseteq \mathbb{R}^p$, let $\phi(\cdot; \mathcal{W}) : \mathcal{X} \rightarrow \mathcal{F}$ be a neural network with $L$ hidden layers and a set of weights $\mathcal{W} = \{W^1, ..., W^L\}$ where $W^l$ are the weights of layer $l \in \{1, ..., L\}$. Therefore, $\phi(x; \mathcal{W}) \in \mathcal{F}$ is the feature representation of $x \in \mathcal{X}$. Unsupervised deep SVDD jointly learns the feature representation and minimizes the volume of a hypersphere in $\mathcal{F}$. Given the unlabeled training data $D_n = \{x_1, ..., x_n\}$, the soft-boundary objective function is given as

$$
\min_{\mathcal{W}} R^2 + \frac{1}{n} \sum_{i=1}^{n} \max\{0, ||\phi(x_i; \mathcal{W}) - c||^2 - R^2\} + \frac{\lambda}{2} \sum_{l=1}^{L} ||W^l||_F^2.
$$

where $R$ is the radius, $c \in \mathcal{F}$ is the center of the hypersphere, and $\nu \in (0, 1]$ is the trade-off between the volume and boundary violations of the hypersphere, i.e. fraction of outliers. $c$ is fixed as the mean of the mapped data from an initial forward pass [13]. For every iteration, $R$ is computed as the $(1 - \nu)$-th quantile of $||\phi(x_i; \mathcal{W}) - c||_2$. Assuming most of the training data is normal, objective (1) can be simplified to the following one-class objective function

$$
\min_{\mathcal{W}} \frac{1}{n} \sum_{i=1}^{n} ||\phi(x_i; \mathcal{W}) - c||^2 + \frac{\lambda}{2} \sum_{l=1}^{L} ||W^l||_F^2.
$$

(2)
B. Semi-supervised Deep SVDD

In the semi-supervised approach, the model has access to both unlabeled training samples \(x_1, \ldots, x_n \in \mathcal{X}\), and labeled training samples \((\tilde{x}_1, \tilde{y}_1), \ldots, (\tilde{x}_m, \tilde{y}_m) \in \mathcal{X} \times \mathcal{Y}\), where \(\mathcal{X} \subseteq \mathbb{R}^d\) and \(\mathcal{Y} = \{-1, +1\}\). Normal samples are labeled as \(\tilde{y} = +1\) and anomalies as \(\tilde{y} = -1\). The one-class semi-supervised Deep SVDD objective function is formulated as

\[
\min_{W} \frac{1}{n + m} \sum_{i=1}^{n} ||\phi(x_i; \mathcal{W}) - c||^2 + \frac{\eta}{n + m} \sum_{j=1}^{m} (||\phi(\tilde{x}_j; \mathcal{W}) - c||^2) \tilde{y}_j. \tag{3}
\]

We omitted the weight decay regularizer in the above formulation for simplicity. For more information, we refer the reader to the work [13], [14].

IV. PROPOSED METHODOLOGY

The hypersphere collapse phenomenon and the architectural constraints for deep SVDD are briefly described in Section IV-A. The two different regularization methods are presented in Section IV-B and Section IV-C, respectively. Section IV-D discusses the adaptive weighting scheme used to control the scaling between the deep SVDD loss and the respective regularizer.

A. Hypersphere Collapse

According to [13], [14], an improperly formulated deep SVDD network (which includes the hypersphere center \(c\)), may lead to the learning of a trivial solution, i.e., hypersphere/mode collapse, a phenomenon where the features learned by the network converge to a single point, \(\phi \equiv c\). It has been recommended that the center \(c\) must be something other than the all-zero-weights solution, and the network should use only unbounded activations and omit bias terms to avoid a hypersphere collapse. However, omitting bias terms in a network may only lead to a sub-optimal feature representation due to the role of bias in shifting activation values. In this work, we propose two simple regularization techniques against mode collapse in deep SVDD which allow the revoke of previous restrictions on the class of networks.

B. Regularization with Random Noise Injection via the Loss Layer

We revisit the timeless idea of noise injection in machine learning, and, inspired by methods which address the overfitting issue in convolutional neural networks (CNNs) [27], [28], we took a similar regularization approach for a loss layer to prevent mode collapse in deep SVDD. We added a new loss layer on top of the feature extractor to learn \(k\) randomly labeled classification tasks in conjunction with the SVDD objective function. For some input space \(\mathcal{X} \subseteq \mathbb{R}^d\), feature space \(\mathcal{F} \subseteq \mathbb{R}^p\), and output space \(\mathcal{K} \subseteq \mathbb{R}^k\), let \(\phi(\cdot; \mathcal{W}_0) : \mathcal{X} \to \mathcal{F}\) be a neural network with \(L\) hidden layers, and \(g(\cdot; \mathcal{W}_1) : \mathcal{F} \to \mathcal{K}\) be the linear layer after it. The final network is given by \(z(x; \mathcal{W}) = g(\phi(x; \mathcal{W}_0); \mathcal{W}_1) \in \mathcal{K}\), where \(\phi(x; \mathcal{W}_0) \in \mathcal{F}\) is the feature representation of \(x \in \mathcal{X}\). For a given input sample \(x_i\), we define \(z_{i,q}\) as the \(q\)-th element of the \(k\)-dimensional logits vector \(z(x_i; \mathcal{W})\) and \(\sigma\) as the sigmoid function. We add the following sigmoid cross-entropy loss as a regularizer to the standard unsupervised or semi-supervised deep SVDD loss \(L_{SVDD}\) (Eq. (1), (2) or (3)).

\[
L_{reg} = -\frac{1}{nk} \sum_{i=1}^{n} \sum_{q=1}^{k} y_{rnd} \log \sigma(z_{i,q}) + (1 - y_{rnd}) \log(1 - \sigma(z_{i,q})). \tag{4}
\]

This approach is based on the idea that learning a single classification problem with meaningful, non-random labels would prevent mode collapse orthogonal to the decision boundary. Therefore, when given random binary labels, the classifier aims at preventing mode collapse along a randomly varying set of hyperplanes, which may potentially lead to better convergence. Here, at each iteration, the binary labels \(y_{rnd}\) are randomly sampled from the uniform distribution of \(\{0, 1\}\) for each sample. At test time, the linear layer \(g(\cdot; \mathcal{W}_1)\) will be removed. We present the overall loss function used in the training of the regularized deep SVDD in Section IV-D.

C. Regularization with Hinge Loss on Minibatch Variance

We also consider using the hinge loss [29] to penalize the minibatch variance if it falls below a specified threshold. This ensures the features in a minibatch to differ sufficiently, thereby preventing mode collapse. Being able to specify the threshold also gives us more control over the regularization as compared to the method presented in Section IV-B which is based on random noise injection. Using the notations from Section IV-B, we define \(\phi_{i,q}\) as the \(q\)-th feature from the feature vector \(\phi(x_i; \mathcal{W}_0) \in \mathcal{F}\) of the input sample \(x_i\). We do not add a linear layer after the feature extractor \(\phi(\cdot; \mathcal{W}_0)\) as done in Section IV-B. We add the following regularization loss to the standard unsupervised or semi-supervised deep SVDD loss \(L_{SVDD}\) (Eq. (1), (2) or (3)).

\[
L_{reg} = \max\{0, d_t - \frac{1}{p(n - 1)} \sum_{q=1}^{p} \sum_{i=1}^{n} (\phi_{i,q} - \bar{\phi}_q)^2\} \tag{5}
\]

\[
= \max\{0, d_t - \frac{1}{p(n - 1)} \sum_{q=1}^{p} \sum_{i=1}^{n} (\phi_{i,q} - \frac{1}{n} \sum_{k=1}^{n} \phi_{k,q})^2\}.
\]

The term \(\frac{1}{p(n - 1)} \sum_{q=1}^{p} \sum_{i=1}^{n} (\phi_{i,q} - \bar{\phi}_q)^2\) computes the sample variance of the minibatch of features averaged over \(p\) dimensions (the dimension of the feature vector) and \(d_t\) refers to the variance threshold used for penalization at epoch \(t\). This formulation enforces a penalty only when the minibatch variance falls below the threshold, i.e. \(\frac{1}{p(n - 1)} \sum_{q=1}^{p} \sum_{i=1}^{n} (\phi_{i,q} - \bar{\phi}_q)^2 < d_t\).

We circumvent the general problem of selecting an appropriate variance threshold with an annealing approach that gradually decreases the threshold \(d_t\) by a factor of 10 every \(r\) epochs.
The overall loss function used in the training of the regularized deep SVDD will be discussed in the next subsection.

**D. Adaptive Weighting Scheme**

We minimize the following objective function for the regularized variant of deep SVDD.

\[ L_{\text{final}} = L_{SVDD} + c_t \cdot L_{\text{reg}}, \quad (6) \]

where \( c_t \) is the adaptive weighted constant defined as follows.

\[ c_t = \alpha c_{t-1} + \beta (1 - \alpha) \frac{L_{SVDD}(t)}{L_{\text{reg}}(t)} \quad (7) \]

At every iteration \( t \), \( c_t \) is updated based on the value of its previous iteration \( c_{t-1} \) and the ratio of the SVDD loss \( L_{SVDD}^{(t)} \) and regularization loss \( L_{\text{reg}}^{(t)} \) from the current minibatch. The contribution of the different components to \( c_t \) is determined by the hyperparameter \( \alpha \in [0, 1] \), while the hyperparameter \( \beta \in [0, 1] \) controls the ratio of the two losses. This adaptive weighting scheme prevents the SVDD loss from dominating the overall loss (6) in the early stages of training. Moreover, it allows an appropriate amount of regularization to be applied without harshly impeding the learning of feature representations. We consider one regularization method at a time and not both methods jointly as we found them not to improve if applied together. Thus, \( L_{\text{reg}} \) refers to either the loss of the regularizer in Section IV-B or IV-C. For the regularizer in Section IV-B, the loss is expected to converge to \( \mathbb{E}[L_{\text{reg}}] = \left(-\frac{1}{2} \log \frac{1}{2} + \frac{1}{2} \log \frac{1}{\alpha} \right) = \log 2 \), as the linear layer acts essentially as a random multi-label classifier trained with binary labels sampled from the uniform discrete distribution. We observe this convergence in our experiments.

**V. EXPERIMENTAL SETTINGS**

**A. Datasets**

We used the CIFAR10 [30] dataset and the more challenging VOC2012 [31] and WikiArt [32], [33] datasets in this work. For reasonable computational speed, the VOC2012 and WikiArt images were resized to be on the same scale as the input samples used in pretrained CNNs which is 224 × 224. For the CIFAR10 and VOC2012 datasets, all the classes were included. For the WikiArt dataset, we included only six styles, i.e. *Art Nouveau Modern, Contemporary Realism, Fauvism, High Renaissance, Naive Art Primitivism,* and *Symbolism* to save training time. We refer to these original, non-manipulated images as natural images. We considered two different experimental setups: the common synthetic one-vs-all setup [13], [22] and our novel setup with image alteration anomalies. In the common setup, only a single class is considered normal at a time, and the remaining \( C - 1 \) classes are considered anomalous. We followed the experimental setups in [13], [14]. For the CIFAR10 dataset, we used 80% of the normal class samples from the given train set as training samples, and the remaining samples in the given train set were used as the validation set for unsupervised learning. For semi-supervised learning, we used 80% of the normal samples from the given train set and 10% of samples from a single anomaly class as training samples. The remaining samples were used for validation. Performance was reported on the given test set.

In our novel setup with image alteration anomalies, we pooled over all natural samples from \( C \) classes and considered them as the normal sample set. We trained only a single anomaly detector using only natural samples from all the \( C \) classes. To create the anomalies, we explored two different methods. The first method involved distorting the natural image by splitting the image into \( N \times N \) blocks and permuting these blocks. The second method involved drawing with certain thickness \( M \), a number of random strokes (one random color per image) of different lengths on various parts of the natural image. Larger values of \( N \) and \( M \) indicate a larger manipulation/distortion of the natural image. The anomalies created from the first method have apparent geometrical structures while the anomalies from the second method do not. We created an anomalous counterpart for each natural sample. For the CIFAR10 dataset, we split 20% of the given train set for validation. We obtained 20,000 samples for both validation and test, and 40,000 training samples for unsupervised learning. For the VOC2012 dataset, we split 60% of the given validation set for test. We obtained 4658 and 6988 validation and test samples respectively, 5717 training samples for unsupervised learning, and 6352 training samples for semi-supervised learning. For the Wikiart dataset, 20% of the given train set was split as validation and the given validation set was used as test set. We obtained 3928 and 8410 validation and test samples respectively, 7856 training samples for unsupervised learning, and 8728 training samples for semi-supervised learning. All train sets for unsupervised learning consist of normal samples only, while the train sets for semi-supervised learning consist of 90% unlabeled samples and 10% labeled anomalies. There are equal number of anomalies and normal samples in both validation and test sets. Our novel setup is more challenging and representative of normal samples encountered in the wild as compared to the one-vs-all setup. AD on the one-vs-all setup which assumes the normal samples come only from a single class may be too easy for datasets with strong geometric shape priors. This will be discussed in Section VI. In addition, our setup adds to the variety of benchmarks and allows us to work on actual anomaly detection problems, as opposed to problems for which specific prior knowledge is available but are treated generically as anomaly detection problems [34], [35]. We used the area under the curve (AUC) of the receiver operating curve as a performance metric. Unless otherwise stated, all reported results are the average of five runs, where each run was initialized with a random seed.

**B. Model Hyperparameters**

We used ResNet-18 [36] as the underlying network architecture with max-pooling layer. For the deep SVDD model and its variants, we finetuned the pretrained ResNet-18 network as opposed to [13], [14] which trained an autoencoder for network initialization. We used the Adam optimizer [37] with
\( \lambda = 5 \cdot 10^{-4} \) and a batch size of 50. We set 30 as the maximum number of epochs and selected the best epoch model based on the validation performance. For the proposed regularized variants, we set \( c_0 = 0, c = 0.9 \), and \( \beta = 0.5 \). For the method in Section IV-B, the optimal number of k classification tasks to use was determined by selecting \( k \) from \( k \in \{5, 30, 50, 60, 70, 100, 130, 200\} \) via grid search using the validation performance as a selection criterion. We used \( r = 3 \) and \( d_0 = 0.1 \) for the method in Section IV-C. A non-decaying learning rate was used to fully analyse the impact of mode collapse. We used a lower learning rate for the other layers as compared to the linear layer collapse. We used a lower learning rate for the other layers learning rate was used to fully analyse the impact of mode collapse. We selected the same set of different methods on the two setups (refer Section V-A). In addition, we explain the stability and performance at different learning rates and discuss the impact of the pooling layer on performance. Lastly, we also extend our experiments to include the semi-supervised deep SVDD. Note that in all tables, we refer to the noise-regularized variant described in Section IV-B as \( D. \text{SVDD w/ bias & noise reg.} \) and the variance-regularized method described in Section IV-C as \( D. \text{SVDD w/ bias & } \sigma^2 \text{ reg.} \). To study the effect of the proposed regularizers on performance, we also include the comparison with a non-regularized variant of deep SVDD with the bias term, i.e. \( D. \text{SVDD w/ bias} \) as shown in Table I, it can be observed that the adaptive weighting scheme generally improves performance in a majority of the settings. Therefore, for the remaining experiments, we use only the adaptive weighting scheme and refrain from using the fixed weighting scheme.

**VI. RESULTS AND DISCUSSION**

In this section, we study the effect of the adaptive weighting scheme on the regularizers, followed by a comparison of the different methods on the two setups (refer Section V-A). In addition, we explain the stability and performance at different learning rates and discuss the impact of the pooling layer on performance. Lastly, we also extend our experiments to include the semi-supervised deep SVDD. Note that in all tables, we refer to the noise-regularized variant described in Section IV-B as \( D. \text{SVDD w/ bias & noise reg.} \) and the variance-regularized method described in Section IV-C as \( D. \text{SVDD w/ bias & } \sigma^2 \text{ reg.} \). To study the effect of the proposed regularizers on performance, we also include the comparison with a non-regularized variant of deep SVDD with the bias term, i.e. \( D. \text{SVDD w/ bias} \). This variant which uses the bias term is prone to mode collapse and is not to be confused with the standard deep SVDD model [13] that follows a set of architectural constraints. We refer to the standard deep SVDD model as \( D. \text{SVDD} \) in the tables. In Table I, it can be observed that the adaptive weighting scheme generally improves performance in a majority of the settings. Therefore, for the remaining experiments, we use only the adaptive weighting scheme and refrain from using the fixed weighting scheme.

**TABLE I**

<table>
<thead>
<tr>
<th>Models</th>
<th>Adaptive Weighting</th>
<th>VOC2012 Dataset</th>
<th>WikiArt Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(proposed)</td>
<td>Strokes 5</td>
<td>Strokes 9</td>
</tr>
<tr>
<td>D. SVDD w/ bias &amp; noise reg.</td>
<td>×</td>
<td>75.6±1.1</td>
<td>76.2±4.4</td>
</tr>
<tr>
<td>D. SVDD w/ bias &amp; ( \sigma^2 ) reg.</td>
<td>✓</td>
<td>76.6±1.8</td>
<td>78.8±1.0</td>
</tr>
</tbody>
</table>

**TABLE II**

<table>
<thead>
<tr>
<th>Models</th>
<th>VOC2012 Dataset</th>
<th>WikiArt Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Block perm.</td>
<td>Block perm.</td>
</tr>
<tr>
<td></td>
<td>2 × 2</td>
<td>8 × 8</td>
</tr>
<tr>
<td>OC-SVM w/ PCA</td>
<td>59.8±0.0</td>
<td>59.8±0.0</td>
</tr>
<tr>
<td>OC-SVM w/ CNN feat.</td>
<td>53.5±0.0</td>
<td>52.9±0.0</td>
</tr>
<tr>
<td>IF w/ PCA</td>
<td>59.6±0.3</td>
<td>82.8±0.2</td>
</tr>
<tr>
<td>IF w/ CNN feat.</td>
<td>51.7±0.7</td>
<td>30.8±3.8</td>
</tr>
<tr>
<td>GEO w/ Dirichlet [22]</td>
<td>78.8±0.5</td>
<td>87.4±1.1</td>
</tr>
</tbody>
</table>

**TABLE III**

<table>
<thead>
<tr>
<th>Models</th>
<th>CIFAR10 Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>One-vs-all</td>
</tr>
<tr>
<td>OC-SVM w/ PCA</td>
<td>61.6±10.4</td>
</tr>
<tr>
<td>OC-SVM w/ CNN feat.</td>
<td>75.3±8.0</td>
</tr>
<tr>
<td>IF w/ PCA</td>
<td>54.5±12.6</td>
</tr>
<tr>
<td>IF w/ CNN feat.</td>
<td>69.2±10.0</td>
</tr>
<tr>
<td>GEO w/ Dirichlet [22]</td>
<td>87.2±7.2</td>
</tr>
<tr>
<td>D. SVDD [13]</td>
<td>67.9±6.5</td>
</tr>
<tr>
<td>D. SVDD w/ bias</td>
<td>67.7±6.1</td>
</tr>
<tr>
<td>D. SVDD w/ bias &amp; noise reg. (proposed)</td>
<td>68.6±6.4</td>
</tr>
<tr>
<td>D. SVDD w/ bias &amp; ( \sigma^2 ) reg. (proposed)</td>
<td>67.6±5.2</td>
</tr>
</tbody>
</table>
also be observed for the one-vs-all setup in Table III. The GEO method does not perform well on the strokes test sets due to apparent geometrical structure such as the strokes anomalies created using method 2 in Section V-A. We elaborate on this later with reference to Table III and IV. In Table II, we also note that at least one of the proposed regularized variants of deep SVDD (mostly the noise-regularized variant, i.e. D. SVDD w/ bias & noise reg.) has the second best performance after the GEO method for a majority of the block permutation settings. In Table III, the OC-SVM model with CNN features performs second best on the one-vs-all setup. Additionally, we observe that the noise-regularized model appears to be more effective than the variance-regularized model in a majority of the cases. For some experiments, we also observe that the shallow models with features from PCA yield better performance than features from ResNet-18, which is partially due to the pooling operation in the pretrained model that has removed some useful features of the anomalies. This will be shown in Figure 2 and discussed later. We also note that the performance achieved by the different models for the one-vs-all setup on CIFAR10 in Table III is similar to other works [13], [22].

The GEO method does not perform well on the strokes test sets as shown in Table III and IV. Since there is no apparent geometrical structure in the strokes test sets due to the various shapes and lengths of the strokes, the GEO model can not take advantage of the geometrical structure of the anomalies and does not perform well in this setting. This suggests that the use of the GEO model may not be appropriate for other types of anomalies that do not portray any apparent geometrical structure. For the shallow models and all variants of deep SVDD, their detections do not directly rely on the

On the block permutation test sets, the GEO model [22] significantly outperforms all the other methods. The same can also be observed for the one-vs-all setup in Table III. The anomalies in the block permutation test set and the normal samples (from a single class) in the one-vs-all setup contain specific geometrical structures that are highly distinguishable from their counterparts. Therefore, the GEO model trained to distinguish salient geometrical features of transformed images performs well on these samples due to the presence of significant geometrical structures. However, the GEO model does not perform well given anomalies that do not portray any apparent geometrical structure such as the strokes anomalies created using method 2 in Section V-A. We elaborate on this later with reference to Table III and IV. In Table II, we also note that at least one of the proposed regularized variants of deep SVDD (mostly the noise-regularized variant, i.e. D. SVDD w/ bias & noise reg.) has the second best performance after the GEO method for a majority of the block permutation settings. In Table III, the OC-SVM model with CNN features performs second best on the one-vs-all setup. Additionally, we observe that the noise-regularized model appears to be more effective than the variance-regularized model in a majority of the cases. For some experiments, we also observe that the shallow models with features from PCA yield better performance than features from ResNet-18, which is partially due to the pooling operation in the pretrained model that has removed some useful features of the anomalies. This will be shown in Figure 2 and discussed later. We also note that the performance achieved by the different models for the one-vs-all setup on CIFAR10 in Table III is similar to other works [13], [22].
presence of geometrical structure in the anomalies and hence, can be used generically for anomaly detection regardless of the types of anomalies. Based on Table III and IV, the variance-regularized model performs better than the other models on the VOC2012 dataset for a majority of the settings, while the noise-regularized model is more effective on the CIFAR10 and WikiArt dataset. It can also be observed that the deep SVDD model and its variants perform better on easier settings, i.e. samples with larger distortion of the natural image.

Regarding impact of learning rates, we observe that both regularized and non-regularized deep SVDD models generally perform better at low learning rate (0.001) as compared to high learning rates (0.01 and 0.1). Close examinations on the experiments also reveal that both regularized variants and the standard deep SVDD model [13] collapse to a single point \( \phi \equiv c \) at high learning rates (0.01 and 0.1). For the variance-regularized method, we observe in our preliminary experiments that fixing the variance threshold \( d_1 \) to a sufficiently small constant can effectively prevent mode collapse for high learning rates, but such a stringent threshold can severely hinder feature learning. Thus, we choose to adopt the annealing approach to gradually decrease the variance threshold \( d_1 \) instead. All results reported at high learning rates (0.01 and 0.1) in Figure 1 and Table V, are the test performances of the best performing model before the hypersphere collapses. Although both regularized models and the standard deep SVDD collapse under high learning rates, \( D. \text{ SVDD} \) w/ bias & noise reg. model achieves better performance convergence than the \( D. \text{ SVDD} \) model for a majority of the experiments as shown in Table V. Moreover, even though the \( D. \text{ SVDD} \) w/ bias & \( \sigma^2 \) reg. model is less effective than the \( D. \text{ SVDD} \) w/ bias & noise reg. model, it shows better performance convergence than the \( D. \text{ SVDD} \) model for half of the experiments at high learning rates (0.01 and 0.1). These observations suggest that the regularizers may have curbed the impact of mode collapse to an extent. Figure 1 clearly depicts the trend of the performance difference between the proposed methods and the standard deep SVDD across different learning rates. For the noise-regularized variant of deep SVDD, we observe in general a higher performance gain over the standard deep SVDD model as the learning rate increases. Meanwhile, no obvious trend is shown for the performance difference between the variance-regularized model and the standard deep SVDD model. For all other experiments, we choose the low learning rate of \( 10^{-3} \) due to better performance and stability against mode collapse.

In addition, we remove the pooling layer of ResNet-18 and compare the performance of the models with the results from Table II. In Figure 2, the OC-SVM and IF models generally show better performance when given ResNet-18 features without pooling. For some settings, these shallow models even outperform their counterparts with PCA features (refer Table II). The improvement in performance suggests that the pooling operation may have removed certain useful features for the anomalous samples. Without the pooling layer, the deep SVDD model and its variants consistently perform better on the VOC2012 dataset and worse for the WikiArt dataset.

**TABLE VI**

<table>
<thead>
<tr>
<th>Models</th>
<th>CIFAR10</th>
<th>VOC2012</th>
<th>WikiArt</th>
</tr>
</thead>
<tbody>
<tr>
<td>One</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-vs-</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>all</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( D. \text{ SVDD} )</td>
<td>80.9±6</td>
<td>87.4±6</td>
<td>85.9±6</td>
</tr>
<tr>
<td>( 14 )</td>
<td>94.3±2</td>
<td>86.7±3</td>
<td>82.8±2</td>
</tr>
<tr>
<td>D. \text{ SVDD} w/ bias</td>
<td>80.8±6</td>
<td>85.4±5</td>
<td>80.2±2</td>
</tr>
<tr>
<td>D. \text{ SVDD} w/ bias &amp; ( \sigma^2 ) reg.</td>
<td>81.2±6</td>
<td>87.9±4</td>
<td>85.4±5</td>
</tr>
<tr>
<td>D. \text{ SVDD} w/ noise reg.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D. \text{ SVDD} w/ noise reg.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( Bp. = \text{ Block perm.} )</td>
<td>80.0±6</td>
<td>89.4±4</td>
<td>88.6±4</td>
</tr>
<tr>
<td>( St. = \text{ Strokes} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \triangle \text{train set} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \text{val. set} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \triangle \text{test set} )</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Lastly, we also show in Table VI, the performance of the deep SVDD model and its variants with semi-supervised learning. For the regularized variants, we apply regularization to all labeled and unlabeled samples in a similar manner as the unsupervised regularized variants. For the one-vs-all setup conducted with the CIFAR10 dataset, the \( D. \text{ SVDD} \) w/ bias & noise reg. model has the best performance. For the image alteration anomalies setup conducted only with the VOC2012 and WikiArt datasets, we ensure that the models are not trained and tested with the same type of anomalies to study the generalization capability of the models to a different type of anomaly. In the first two columns for VOC2012 and WikiArt datasets, the models were trained and validated/tested with different type of anomalies (block permutation vs. strokes). It can be seen that the models trained with block permutation train set show better performance on strokes test set, as compared to models trained with strokes train set and tested on block permutation test sets. Similar to unsupervised learning, the semi-supervised noise-regularized variant generally achieves
better performance than the variance-regularized variant. For almost all settings, it can be observed that the proposed regularizers (mostly D. SVDD w/ bias & noise reg.) perform notably better than the other deep SVDD models.

VII. CONCLUSION

In this work, we considered two simple and effective regularization methods against mode collapse in deep SVDD. Even though the minibatch variance regularization method provided more control over the amount of regularization applied, the regularization method based on random noise injection generally performed better. Additionally, we demonstrated that both regularization methods performed better than the state-of-the-art GEO method on the strokes test sets that do not portray apparent geometrical structures. We also showed that either one of the regularized variants achieved better performance than the standard deep SVDD and other classification-based models in a majority of the experiments. Moreover, we learned that the standard deep SVDD model and its variants performed better and were more stable at low learning rate. As noted by one of the reviewers, our proposed regularizers could be of interest for mitigating the mode collapse problem in GANs.

ACKNOWLEDGMENT

PC acknowledges support by the ST Engineering Electronics-SUTD Cyber Security Laboratory. AB acknowledges support by the ST Engineering Electronics-SUTD Cyber Security Laboratory and Ministry of Education of Singapore (MoE) Tier 2 grant MOE2016-T2-2-154. LR acknowledges support by the German Ministry of Education and Research (BMBF) in the project ALICE III (01IS18049B). MK acknowledges support by the German Research Foundation (DFG) award KL 2698/2-1 and by the Federal Ministry of Science and Education (BMBF) awards 031L0023A, 01IS18051A, and 031B0770E.

REFERENCES


