Received 28 December 2023, accepted 11 January 2024, date of publication 16 January 2024, date of current version 24 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3355021

RESEARCH ARTICLE

Few-Shot Anomaly Detection via Personalization

SANGKYUNG KWAK[1], (Member, IEEE), JONGHEON JEONG[1], HANKOOK LEE[2], WOOHYUCK KIM[1], DONGHO SEO[3], WOOJIN YUN[3], WONJIN LEE[3], AND JINWOO SHIN[1]

[1]Kim Jaechul Graduate School of AI, Korea Advanced Institute of Science and Technology, Daejeon 34141, Republic of Korea
[2]LG AI Research, Seoul 07796, Republic of Korea
[3]LIG Nex1, Geonggi 13488, Republic of Korea

Corresponding author: Sangkyung Kwak (skkwak9806@kaist.ac.kr)

This work was supported by the Korea Research Institute for Defense Technology Planning and Advancement-Grant funded by the Defense Acquisition Program Administration (DAPA) under Grant KRIT-CT-21-033.

ABSTRACT Even with a plenty amount of normal samples, anomaly detection has been considered as a challenging machine learning task due to its one-class nature, *i.e.*, the lack of anomalous samples in training time. It is only recently that a *few-shot* regime of anomaly detection became feasible in this regard, *e.g.*, with a help from large vision-language pre-trained models such as CLIP, despite its wide applicability. In this paper, we explore the potential of large *text-to-image generative models* in performing few-shot industrial anomaly detection. Specifically, recent text-to-image models have shown unprecedented ability to generalize from few images to extract their common and unique concepts, and even encode them into a textual token to ''personalize'' the model: so-called *textual inversion*. Here, we question whether this personalization is specific enough to discriminate the given images from their potential anomalies, which are often, *e.g.*, open-ended, local, and hard-to-detect. We observe that standard textual inversion exhibits a weaker understanding in localized details within objects, which is not enough for detecting industrial anomalies accurately. Thus, we explore the utilization of model personalization to address anomaly detection and propose Anomaly Detection via Personalization (ADP). ADP enables extracting fine-grained local details shared in the images with simple-yet an effective regularization scheme from the zero-shot transferability of CLIP. We also propose a self-tuning scheme to further optimize the performance of our detection pipeline, leveraging synthetic data generated from the personalized generative model. Our experiments show that the proposed inversion scheme could achieve state-of-the-art results on two industrial anomaly benchmarks, MVTec-AD and VisA, in the regime of few normal samples.

INDEX TERMS Industrial anomaly detection, model personalization, text-to-image diffusion model, vision-language model.

I. INTRODUCTION

The ability to identify unusual patterns in images is a natural capability of human cognition. Even when provided with only a small number of normal examples, humans can adapt to discriminate abnormality from the examples, whereas this remains a challenging task in the field of computer vision. *Anomaly detection* (AD), where the task is formulated, faces fundamental challenges due to several reasons. Firstly, objects and their defects can vary widely in terms of color, texture, and size across numerous industrial domains: *e.g.*, aerospace, automobiles, pharmaceuticals, and electronics. Besides, some types of anomaly can be fine-grained which has only little differences between normal and anomalous data while other can be coarse-grained. Secondly, obtaining and specifying the expected variations in defects is limited and costly in real-world situations.

Upon these fundamental challenges, significant efforts have been made to approach AD: especially in *one-class*, semi-supervised setting [1], [2], [3], [4], [5], [6], [7], or in self-supervised setting [8], to name a few. Intuitively, the major technical bottleneck here is to learn features expressive

The associate editor coordinating the review of this manuscript and approving it for publication was Chengpeng Hao.

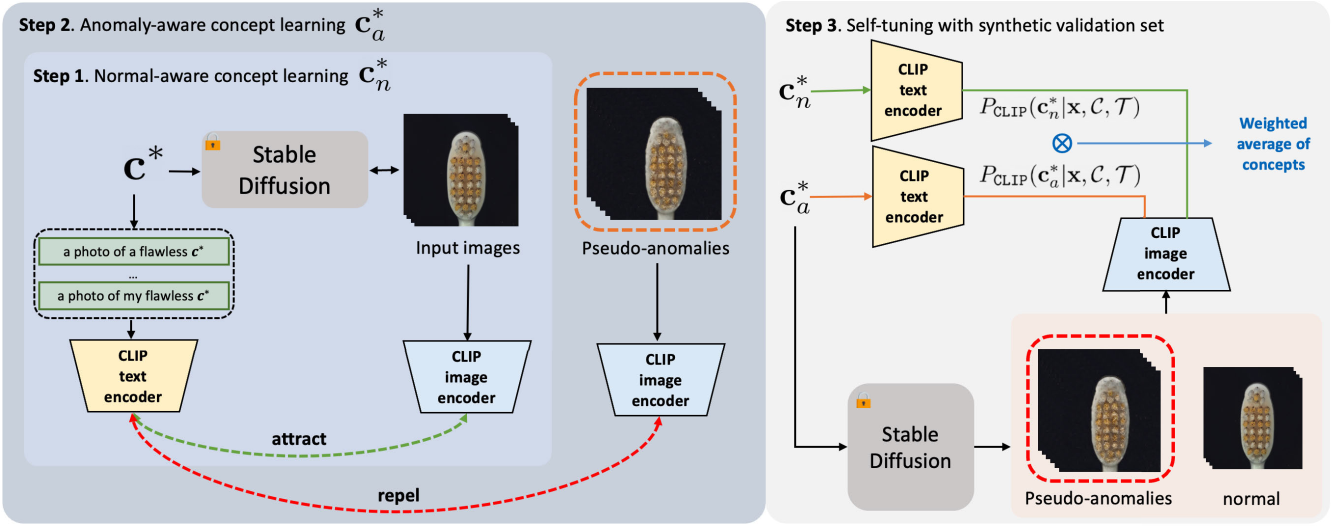
Step 2. Anomaly-aware concept learning $\mathbf{c}_a^*$
Step 1. Normal-aware concept learning $\mathbf{c}_n^*$
$\mathbf{c}^*$
Stable Diffusion
Input images
Pseudo-anomalies
a photo of a flawless c^*
...
a photo of my flawless c^*
CLIP text encoder
CLIP image encoder
CLIP image encoder
attract
repel
Step 3. Self-tuning with synthetic validation set
$\mathbf{c}_n^*$
$\mathbf{c}_a^*$
CLIP text encoder
CLIP text encoder
$P_{\texttt{CLIP}}(\mathbf{c}_n^*|\mathbf{x}, \mathcal{C}, \mathcal{T})$
$\otimes$
$P_{\texttt{CLIP}}(\mathbf{c}_a^*|\mathbf{x}, \mathcal{C}, \mathcal{T})$
Weighted average of concepts
CLIP image encoder
Stable Diffusion
Pseudo-anomalies
normal

FIGURE 1. Overview of *Anomaly Detection via Personalization* (ADP). In Step 1, normal concepts are converted into $\mathbf{c}_n^*$ by guiding the normal prompt incorporating $\mathbf{c}_n$ to be closer to the given images (Section IV-A). In Step 2, anomalous concepts are converted into $\mathbf{c}_a^*$ by additionally distancing pseudo-anomalies to the normal prompt incorporating $\mathbf{c}_a$ (Section IV-B). In Step 3, by utilizing CLIP, the use of $\mathbf{c}_n^*$ and $\mathbf{c}_a^*$ are further tuned with synthesized pseudo-anomalies (Section IV-C).

enough to encode inter-class variability without knowing anomalous data, while maintaining intra-class variability induced from normal data. To overcome this, there have been two representative approaches: (a) *feature-based approaches* [2], [3], [6], [9], [10], [11], [12] leverage an external, pre-trained feature extractor, *e.g.*, on ImageNet, to retrieve its richer features in modeling AD; and (b) *reconstruction-based approaches* [5], [7], [13], [14], [15] instead model a generative model to extract faithful features in the normal data available, in an attempt to improve the sensitivity of features. With hundreds to thousands of normal images, such approaches have shown effectiveness to achieve high-enough detection performances, *e.g.*, on existing industrial anomaly detection benchmarks [16], [17].

Anomaly detection with limited data, *e.g.*, with only *few normal images*, has been still challenging even until recently. The cost-efficiency of a *language-driven prior* has emerged as an effective way to mitigate the challenge, particularly since CLIP [18], a recent large vision-language model. For example, Jeong et al. [19] have demonstrated state-of-the-art performances in few-shot AD by incorporating a "zero-shot", language-driven AD pipeline from CLIP, *e.g.*, by additionally comparing similarities to words "normal" *vs.* "damaged" for a given image: a similar exploration has been made in the context of novelty detection (or so-called out-of-distribution detection) by Ming et al. [20]. Although it is evident that language can be a useful prior for AD, *e.g.*, to clarify the vague concepts of abnormality by supplying label words (*e.g.*, bottle, capsule, *etc.*), the current interface of "hand-crafting" language prompts becomes a limiting bottleneck as the given AD task gets more specific to the (few-shot) data: and accordingly as it gets "harder-to-describe". In turn, it is observed that the performance of current language-based AD is highly dependent by the prompt design, which is heuristic in nature and requires a careful tuning by humans. For example, Jeong et al. [19] indeed assumed the knowledge of class labels as text words in performing their zero-/few-shot AD.

Contribution: In this paper, we propose a new design of language-based AD, coined *Anomaly Detection via Personalization* (ADP), which leverages *model personalization* [21], [22], [23] that is recently enabled by large-scale text-to-image generative models [24], [25]. Specifically, recent text-to-image generative models have shown capabilities to extract detailed concepts shared across a few given images, and encode them as a *textual token* to compose natural language sentences associated with the generative model: it can "personalize" the model to generate images containing the concepts. Here, we focus on exploring whether this new ability of *textual inversion* could replace the current brittleness in crafting few-shot, language-based AD in practice. We first observe that the current objective for textual inversion (in the context of generative modeling) may not be specific enough to perform accurate few-shot AD. Motivated by this, we propose a novel textual inversion scheme to improve its specificity, based on a richer guidance induced by CLIP [18]. We develop a two-step inversion scheme designed for general AD: the former to personalize from normal samples, and the latter to refine itself based on the personalized model, particularly leveraging the "synthetic" anomaly samples that the model can generate. In this way, the inversion can better capture fine-grained visual semantics which is demanded to perform an accurate AD. We also propose to re-utilize the anomaly synthesis scheme for a self-tuning of our AD model, which is a unique ability to our framework.

With the proposed method, we tackle *few-normal-shot* AD, *viz.*, 2 to 16, an under-explored setup due to its difficulty [8], [26], [27]. We summarize our main contributions in what follows:

- We introduce a novel method to capture unique concepts of anomalies into the token, which improves few-shot AD.
- Using the anomaly-aware token, we show that we can effectively synthesize pseudo-anomalies with pre-trained text-to-image diffusion model.
- We propose a simple yet effective self-tuning method to utilize the tokens in the pre-trained vision-language model for AD.
- Through an extensive evaluation on MVTec-AD and VisA, we report new state-of-the-art results on few-shot AD, *e.g.*, **97.1%** on MVTec-AD and **89.7%** on VisA in AUROC in 16-shot AD, notably even without text descriptions on the object labels as assumed in prior art [19].

II. RELATED WORK

A. ANOMALY DETECTION

In the field of anomaly detection, the focus has been on one-class methods that utilize a large amount of normal images [2], [3], [4], [7], [10], [28]. Specifically, in industrial anomaly detection, which requires to learn unique nominal features, recent works suggest utilizing pre-trained models with external image dataset [2], [3]. However, these existing approaches encounter limitations when applied to specific applications due to the challenges posed by the full-normal-shot setup in MVTec-AD benchmark [16]. Recent studies [8], [26] have investigated few-shot setups by employing augmentation techniques to expand the small support set, leading to enhanced modeling of normality. Another approach, RegAD [27], introduces the concept of model re-using which pre-trains an object-agnostic registration network with diverse images to establish normality for unseen objects. Additionally, utilizing pre-trained vision-language model to extract the prior knowledge has shown remarkable improvement in few-normal-shot anomaly detection [19]. The few-shot setups in anomaly detection is still under-explored and has room for improvement.

B. TEXT-TO-IMAGE DIFFUSION MODELS

At a high level, diffusion models [29], [30], class of generative models, learn the target distribution $p_{\text{data}}(\mathbf{x})$ by learning a gradual denoising process from Gaussian prior distribution to reach $p_{\text{data}}(\mathbf{x})$. The field of diffusion models has seen a wide range of applications, including text-to-image generation. Text-to-image diffusion models are able to generate images conditioned by text prompts [24], [25], [31], which show promising result in image synthesis. Among them, one notable approach is Stable Diffusion [24], which is a popular variant of latent diffusion models (LDMs) [24]. This is trained on extremely large-scale data, have demonstrated remarkable generalization ability. To utilize the strong generalizability in synthesizing images, we incorporate Stable Diffusion to address anomaly detection task.

C. PERSONALIZATION OF TEXT-TO-IMAGE MODELS

With the outstanding scalability of pre-trained text-to-image diffusion models, recent works make great efforts to generate specific instances like personal animals or rare categories. To inject the new concept to the pre-trained models while preserving the previous knowledge, recent works suggest several approaches. This includes fine-tuning only subset of the parameters [23], fine-tuning with the method to preserve prior knowledge [22] and introducing and optimizing a word vector for the new concept [21]. In this way, models excel at integrating new information into their domain without forgetting the prior or overfitting to a small subset of training images. Motivated from this, we suggest utilizing model personalization in identifying anomalies, which enables addressing few-shot setting in anomaly detection task.

III. PRELIMINARIES

A. PROBLEM SETUP

Anomaly detection (AD) aims to determine the presence of "abnormality" given an image $\mathbf{x} \in \mathcal{X}$. We formulate AD as a binary classification problem $\mathcal{X} \to \{0, 1\}$, where "1" indicates the presence of abnormality. Due to the lack of anomalous samples in practice, AD is often assumed to be *one-class*, *i.e.*, its training data $\mathcal{D} := \{(x_i, 0)\}_{i=1}^{K}$ consists of only normal (or negative) samples. In this work, we follow this one-class protocol, particularly focusing on *extreme few-shot* scenarios where the training data only consists of $K = 2$ to 16 normal images. It is also a practice to cast AD as a problem of assigning *anomaly score* rather than a direct classification, again due to the high-imbalance in data: the actual classification in practice is done by thresholding the score.

To solve this extreme few-shot AD task, we utilize vision-language foundation models, a contrastive encoder (*e.g.*, CLIP [18]) and a diffusion model (*e.g.*, LDMs [24]), pre-trained on external datasets. Our approach is widely applicable as the foundation models have shown to be generalizable across various downstream tasks and they are publicly available. We will describe the vision-language contrastive encoder and diffusion model in Section III-B and Section III-C, respectively.

B. CONTRASTIVE LANGUAGE IMAGE PRE-TRAINING

Contrastive language image pre-training (CLIP) [18] is a large-scale pre-training method that offers a joint vision-language representation by training an image encoder $f(\cdot)$ and a text encoder $g(\cdot)$ using contrastive learning [32], [33] with the million-scale image-text pairs from the web. One attractive ability of CLIP is zero-shot transfer, especially for image classification. To be specific, given a set of labels $\mathcal{C} = \{\mathbf{c}_1, \ldots, \mathbf{c}_N\}$, an image $\mathbf{x}$ can be classified by the following probability:

$$P_{\texttt{CLIP}}(\mathbf{c}_i|\mathbf{x}, \mathcal{C}, \mathcal{T}) := \frac{\exp\left(\texttt{sim}\left(f(\mathbf{x}), \mathcal{G}(\mathbf{c}_i)\right)/\tau\right)}{\sum_{j=1}^{N} \exp\left(\texttt{sim}\left(f(\mathbf{x}), \mathcal{G}(\mathbf{c}_j)\right)/\tau\right)}, \tag{1}$$

where $\mathcal{G}(\mathbf{c}_i) = \frac{1}{|\mathcal{T}|}\sum_{T\in\mathcal{T}} g(T(\mathbf{c}_i))$, $\mathtt{sim}(\cdot,\cdot)$ is the cosine similarity, $\tau > 0$ is the temperature hyperparameter, and $T \in \mathcal{T}$ is a prompt template attached to a label $\mathbf{c}$ such as "a photo of a [$\mathbf{c}$]". Note that using multiple templates, *i.e.*, template ensemble, can improve the zero-shot classification accuracy [18].

C. TEXTUAL INVERSION

Textual inversion [21] aims to learn the concept $\mathbf{c}$, *i.e.*, a new pseudo-word describing the common appearance of few images, by directly optimizing the corresponding embedding vector in LDM [24]'s text embedding space which we denote by v. Then, the concept can be composed into new sentences like any other word for a *personalized generation*, i.e., to sample from the distribution $p(\mathbf{x}|\mathbf{c})$. To this end, we use a pre-trained text-to-image latent diffusion model [24], $p_{\mathtt{t2i}}(\mathbf{x}|\mathbf{s})$, where $\mathbf{s}$ is a conditioning text. Given a set of images $\{\mathbf{x}_i\}_{i=1}^K$, the textual inversion finds their common concept $\mathbf{c}^*$ by solving the following optimization problem:

$$v^* := \arg\max_v \sum_{i=1}^{K}\sum_{T\in\mathcal{T}} \log p_{\mathtt{t2i}}(\mathbf{x}_i|T(\mathbf{c})), \tag{2}$$

where $\mathcal{T}$ is a set of prompt templates. After textual inversion, one can generate a new image of the concept $\mathbf{c}^*$ with a template $T \in \mathcal{T}$, *i.e.*, $\mathbf{x} \sim p_{\mathtt{t2i}}(\cdot|T(\mathbf{c}^*))$. Gal et al. [21] found that the concept is well-optimized with only a few images, *e.g.*, $K = 4$. In addition, one can use the text embedding vector of concept on the CLIP representation space by utilizing CLIP text encoder g.

IV. ANOMALY DETECTION VIA PERSONALIZATION

In this section, we introduce Anomaly Detection via Personalization (ADP), a novel framework for few-shot anomaly detection utilizing the ground knowledge in vision-language foundation models. To be specific, ADP finds the concept word that can (i) generate both normal and abnormal images via textual inversion and also (ii) detect the abnormality via CLIP zero-shot classification. In detecting abnormality, ADP combines the complementary prediction by incorporating both concept word and multi-level image features.

To perform anomaly detection (*i.e.*, 2-way classification) without label information, we utilize normal and anomalous state templates, S_n and S_a, respectively, for a concept word $\mathbf{c}$, following Jeong et al. [19]:

$$S_n(\mathbf{c}) := \text{“}\mathtt{flawless}\ [\mathbf{c}]\text{”}, \quad S_a(\mathbf{c}) := \text{“}\mathtt{damaged}\ [\mathbf{c}]\text{”}.$$

To condition the generation and utilize CLIP zero-shot transferability as regularization scheme, we randomly select neutral context texts following Gal et al. [21]. These contain prompts of the form "a photo of a $\mathbf{c}$", "a rendering of a $\mathbf{c}$", etc, by attaching $\mathbf{c}$ to the selected prompt template $T \in \mathcal{T}$. The full list of templates is provided in the Appendix A.

Given a set of a few normal images $\{\mathbf{x}_i\}_{i=1}^K$, the state templates S_n and S_a, and a set of prompt templates $\mathcal{T}$, ADP follows the following procedure:

Step 1. Find the *normal-aware concept* $\mathbf{c}_n^*$ by guiding *normal* images to be close to the *normal* state prompts (Section IV-A).

Step 2. Find the *anomaly-aware concept* $\mathbf{c}_a^*$ by guiding *pseudo-anomalous* images to put distance to the *normal* state prompts (Section IV-B).

Step 3. Perform anomaly detection using the concepts, $\mathbf{c}_n^*$ and $\mathbf{c}_a^*$ (Section IV-C).

A. NORMAL-AWARE CONCEPT LEARNING

In normal-aware concept learning, we aim to capture the visual normal concept $\mathbf{c}_n^*$ from the normal images $\{\mathbf{x}_i\}_{i=1}^K$. To this end, in addition to textual inversion, we make embeddings of the given normal images similar with that of the normal state prompt $T(S_n(\mathbf{c}))$, while dissimilar with that of the anomalous state prompt $T(S_a(\mathbf{c}))$. Normal state prompt $T(S_n(\mathbf{c}))$ and anomalous state prompt $T(S_a(\mathbf{c}))$ can be obtained with attaching concept $\mathbf{c}$ to each state template following randomly selected prompt template, respectively. One example for $T(S_n(\mathbf{c}))$ and $T(S_a(\mathbf{c}))$ are "a photo of a flawless [$\mathbf{c}$]" and "a photo of a damaged [$\mathbf{c}$]". Here, we directly optimize the embedding of concept as described in Section III-C. Formally, the normal-aware concept $\mathbf{c}_n^*$ can be obtained by solving the following optimization problem:

$$\begin{aligned} v_n^* &= \arg\max_v \mathcal{J}_n(\mathbf{c}; \{\mathbf{x}_i\}_{i=1}^K) \\ &:= \arg\max_v \sum_{i=1}^{K}\sum_{T\in\mathcal{T}} \log p_{\mathtt{t2i}}(\mathbf{x}_i|T(S_n(\mathbf{c}))) \\ &\quad + \alpha P_{\mathtt{CLIP}}(S_n(\mathbf{c})|\mathbf{x}_i, \mathcal{C}(\mathbf{c}), \mathcal{T}), \end{aligned} \tag{3}$$

where $\mathcal{C}(\mathbf{c}) = \{S_n(\mathbf{c}), S_a(\mathbf{c})\}$ is the set of normal and anomalous state prompts of the concept $\mathbf{c}$ and α is a hyperparameter. We initialize the embedding of concept $\mathbf{c}$ as that of the word "object" which is applicable to regardless of the domain and dataset.

B. ANOMALY-AWARE CONCEPT LEARNING

We here aim to further capture the "anomalous" concept $\mathbf{c}_a^*$ by integrating synthetic anomalous images. To be specific, we further make embeddings of synthetic anomalous images dissimilar with that of the normal state prompt $T(S_n(\mathbf{c}))$, while maintaining the visual normal concept of the normal images $\{\mathbf{x}_i\}_{i=1}^K$. To this end, we first synthesize *pseudo-anomalous* images via text-guided image manipulation [34] using the text-to-image diffusion model $p_{\mathtt{t2i}}(\mathbf{x}|\mathbf{s})$. We here use a normal image $\mathbf{x}_i$ as a reference image and "a photo with damage" or "a photo of an object with damage" as a conditioning text $\mathbf{s}$. To give more diversity, the manipulated images are further augmented with random resizing and cropping. We denote $\{\tilde{\mathbf{x}}_j\}_{j=1}^L$ as synthesized pseudo-anomalous images (*i.e.*, *pseudo-anomalies*). The examples of pseudo-anomalies are illustrated in the Fig. 3.

In addition to normal-aware concept learning, we put distance between the pseudo-anomalous images $\{\tilde{\mathbf{x}}_j\}_{j=1}^{L}$ and the normal state prompt $T(S_n(\mathbf{c}))$. Formally, the anomaly-aware concept $\mathbf{c}_a^*$ can be obtained by solving the following optimization problem:

$$\begin{aligned} v_a^* &= \arg\max_v \mathcal{J}_a(\mathbf{c}; \{\mathbf{x}_i\}, \{\tilde{\mathbf{x}}_j\}) \\ &:= \arg\max_v \mathcal{J}_n(\mathbf{c}; \{\mathbf{x}_i\}) \\ &\quad - \alpha \sum_{j=1}^{L} \left(P_{\texttt{CLIP}}(S_n(\mathbf{c}) | \tilde{\mathbf{x}}_j, \mathcal{C}(\mathbf{c}), \mathcal{T}) - \gamma \right)^+, \end{aligned} \tag{4}$$

where $(\cdot)^+ := \max(\cdot, 0)$, $\mathcal{C}(\mathbf{c}) = \{S_n(\mathbf{c}), S_a(\mathbf{c})\}$, α and γ are hyperparameters. We initialize the embedding of concept $\mathbf{c}$ as that of the normal-aware concept $\mathbf{c}_n^*$ described in Section IV-A. Since $\mathbf{c}_n^*$ captures high-level visual features of normal images, initializing the concept with the normal-aware helps learning fine-grained anomalous features.

C. ANOMALY DETECTION WITH LEARNED CONCEPTS

We now introduce a simple yet effective detection scheme using the learned concepts. At a high-level, our scheme first extracts CLIP text embeddings of all available prompt state templates with the learned concepts and then mix them to construct 2-way classification prototypes via *self-tuning*. Given a test image, we detect whether it is in-distribution or not using its CLIP image embedding.

1) SELF-TUNING

To utilize both concepts, we mix the concepts using importance weights obtained by a pseudo-validation set, which consists of the normal images $\{\mathbf{x}_i\}_{i=1}^{K}$ and new pseudo-anomalous images $\{\tilde{\mathbf{x}}_j\}_{j=1}^{L'}$ synthesized by conditioning texts, "`a photo of a damaged` $[\mathbf{c}_a^*]$" and "`a photo of a` $[\mathbf{c}_a^*]$ `with damage`", as described in Section IV-B. The importance weight $w(\mathbf{c})$ of each concept $\mathbf{c}$ can be computed by evaluating CLIP zero-shot classification as follows:

$$\begin{aligned} w(\mathbf{c}) &:= \frac{1}{K} \sum_{i=1}^{K} P_{\texttt{CLIP}}(S_n(\mathbf{c}) | \mathbf{x}_i, \mathcal{C}(\mathbf{c}), \mathcal{T}) \\ &\quad + \frac{1}{L'} \sum_{j=1}^{L'} P_{\texttt{CLIP}}(S_a(\mathbf{c}) | \tilde{\mathbf{x}}_j, \mathcal{C}(\mathbf{c}), \mathcal{T}). \end{aligned} \tag{5}$$

Computing importance weight $w(\mathbf{c})$ involves determining the optimal ratio for combining the concepts in performing anomaly detection based on CLIP similarity between images and prompts containing each concept. We then compute the weighted average of the CLIP text embeddings to construct the classification prototype vectors using the CLIP text encoder g as follows:

$$\mathbf{p}_{\texttt{S}} := \frac{1}{|\mathcal{T}|} \sum_{T \in \mathcal{T}} \frac{w(\mathbf{c}_n^*) \cdot g(T(S_{\texttt{S}}(\mathbf{c}_n^*))) + w(\mathbf{c}_a^*) \cdot g(T(S_{\texttt{S}}(\mathbf{c}_a^*)))}{w(\mathbf{c}_n^*) + w(\mathbf{c}_a^*)}. \tag{6}$$

2) ANOMALY DETECTION

Given a test image $\mathbf{x}$, our detection score $\texttt{ADP}(\mathbf{x})$ is formally defined by

$$\texttt{ADP}(\mathbf{x}) := \frac{\exp\left(\texttt{sim}(f(\mathbf{x}), \mathbf{p}_n)/\tau\right)}{\exp\left(\texttt{sim}(f(\mathbf{x}), \mathbf{p}_n)/\tau\right) + \exp\left(\texttt{sim}(f(\mathbf{x}), \mathbf{p}_a)/\tau\right)}. \tag{7}$$

To further improve detection performance, we utilize visual features (*i.e.*, feature maps) using the CLIP image encoder to perform complementary prediction from both language-guided and visual based approaches following Jeong et al. [19]. Specifically, we consider *reference association* module, which enables the storage and retrieval of memory features $\mathbf{R}$ computed from a given set of normal images $\{\mathbf{x}_i\}_{i=1}^{K}$. We further define the feature similarity score by the cosine-similarity to the nearest features in $\mathbf{R}$. For a given dense feature $\mathbf{F} \in \mathbb{R}^{h \times w \times d}$ obtained from a test image, the score is defined as follows:

$$\mathbf{M}_{ij} := \min_{r \in \mathbf{R}} \frac{1}{2}(1 - \langle \mathbf{F}_{ij}, r \rangle). \tag{8}$$

We incorporate three different features are incorporated: small-scale feature $\mathbf{F}^{\texttt{W}}\texttt{s}$, mid-scale feature $\mathbf{F}^{\texttt{W}}\texttt{m}$, and penultimate feature $\mathbf{F}^{\texttt{P}}$. By applying the reference association module, we obtain three reference memories: $\mathbf{R}_{\texttt{s}}^{\texttt{W}}$, $\mathbf{R}_{\texttt{m}}^{\texttt{W}}$, and $\mathbf{R}^{\texttt{P}}$. Then we compute the average of multi-scale prediction (8), and it is given as:

$$\mathbf{M}^{\texttt{W}} := \frac{1}{3}(\mathbf{M}^{\texttt{P}} + \mathbf{M}_{\texttt{s}}^{\texttt{W}} + \mathbf{M}_{\texttt{m}}^{\texttt{W}}). \tag{9}$$

Subsequently, the maximum value of $\mathbf{M}^{\texttt{W}}$ is integrated into the ADP anomaly detection score (7). This score captures complementary information derived from the spatial features of the few-shot references. The complete form of ADP anomaly detection ($\texttt{ADP}_{ad}$) is as follows:

$$\texttt{ADP}(\mathbf{x})_{ad} := \frac{1}{2}\left(\texttt{ADP}(\mathbf{x}) + \max_{ij} \mathbf{M}_{ij}^{\texttt{W}}\right). \tag{10}$$

Remark: (1) Our work differs from (and is complementary to) textual inversion by enabling AD with learned concepts rather than generating personalized images. We show that guidance induced from CLIP plays a unique role in converting image features to the concept which leverages aligning anomalous features into language — improving the separability between *normal* and *anomalous* data. (2) Moreover, ADP is different from WinCLIP [19] in how to incorporate few-shot images into text prompt templates. ADP fully utilizes the few-shot images to extract a shared (language) concept through the personalization technique which allows us to construct meaningful text prompts "without ground-truth labels".

V. EXPERIMENTS

We conduct an extensive evaluation on the proposed method, ADP, on MVTec-AD [16] and VisA [17] benchmarks, two popular datasets in AD capturing real-world scenarios of

TABLE 1. Anomaly detection (AD) performance on MVTec-AD and VisA benchmarks for 2-shot. We report the mean AUROC (%) and standard deviation over three random seeds for each measurement. The results of SPADE, PaDiM and PatchCore are from those reported by Jeong et al. [19].

Data \ Method	SPADE	PaDiM	PatchCore	WinCLIP+	**ADP**	**ADP**$_\ell$
MVTec-AD	$82.9_{\pm 2.6}$	$78.9_{\pm 3.1}$	$86.3_{\pm 3.3}$	$93.8_{\pm 1.0}$	$94.4_{\pm 1.2}$	$\mathbf{95.4}_{\pm 0.9}$
VisA	$80.7_{\pm 5.0}$	$67.4_{\pm 5.1}$	$81.6_{\pm 4.0}$	$84.2_{\pm 0.2}$	$85.7_{\pm 0.9}$	$\mathbf{86.9}_{\pm 0.9}$

TABLE 2. Class-wise comparison of anomaly detection (AD) performance on MVTec-AD benchmark on 4-, 8-, and 16- shots. We report the mean AUROC (%) and standard deviation over three random seeds for each measurement, which highest AUROC (%) for each class is marked as bold.

	4-shot			8-shot			16-shot		
Data \ Method	WinCLIP+	**ADP**	**ADP**$_\ell$	WinCLIP+	**ADP**	**ADP**$_\ell$	WinCLIP+	**ADP**	**ADP**$_\ell$
Bottle	$93.4_{\pm 0.3}$	$\mathbf{98.9}_{\pm 0.4}$	$97.2_{\pm 0.7}$	$93.7_{\pm 0.1}$	$\mathbf{99.4}_{\pm 0.3}$	$97.5_{\pm 1.0}$	$93.7_{\pm 0.2}$	$\mathbf{99.4}_{\pm 0.3}$	$97.5_{\pm 1.1}$
Cable	$83.0_{\pm 0.0}$	$87.9_{\pm 2.5}$	$\mathbf{88.3}_{\pm 3.1}$	$83.0_{\pm 0.1}$	$88.0_{\pm 1.9}$	$\mathbf{88.5}_{\pm 2.4}$	$83.1_{\pm 0.1}$	$\mathbf{88.8}_{\pm 1.1}$	$88.6_{\pm 1.3}$
Capsule	$\mathbf{84.4}_{\pm 9.4}$	$83.4_{\pm 11.9}$	$84.0_{\pm 11.5}$	$90.9_{\pm 1.4}$	$\mathbf{93.1}_{\pm 1.7}$	$93.0_{\pm 1.5}$	$91.7_{\pm 1.5}$	$\mathbf{94.4}_{\pm 2.1}$	$93.5_{\pm 1.0}$
Carpet	$\mathbf{100}_{\pm 0.0}$	$99.9_{\pm 0.1}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$99.5_{\pm 0.7}$	$99.7_{\pm 0.4}$	$\mathbf{100}_{\pm 0.0}$	$99.8_{\pm 0.2}$	$99.9_{\pm 0.1}$
Grid	$99.1_{\pm 0.2}$	$98.0_{\pm 2.5}$	$\mathbf{99.5}_{\pm 0.6}$	$99.0_{\pm 0.5}$	$98.2_{\pm 1.7}$	$\mathbf{99.4}_{\pm 0.4}$	$99.2_{\pm 0.1}$	$98.5_{\pm 2.1}$	$\mathbf{99.4}_{\pm 0.7}$
Hazelnut	$97.5_{\pm 0.1}$	$\mathbf{99.4}_{\pm 0.5}$	$98.9_{\pm 0.4}$	$97.7_{\pm 0.1}$	$\mathbf{99.5}_{\pm 0.7}$	$99.1_{\pm 0.5}$	$97.5_{\pm 0.1}$	$\mathbf{99.5}_{\pm 0.6}$	$99.1_{\pm 0.4}$
Leather	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$
Metal nut	$95.7_{\pm 0.3}$	$99.4_{\pm 0.5}$	$\mathbf{99.6}_{\pm 0.2}$	$95.8_{\pm 0.4}$	$\mathbf{99.6}_{\pm 0.4}$	$\mathbf{99.6}_{\pm 0.2}$	$96.0_{\pm 0.3}$	$\mathbf{99.8}_{\pm 0.1}$	$99.7_{\pm 0.3}$
Pill	$90.1_{\pm 0.1}$	$\mathbf{95.2}_{\pm 0.3}$	$94.9_{\pm 0.6}$	$90.1_{\pm 0.1}$	$\mathbf{95.6}_{\pm 0.5}$	$94.9_{\pm 0.5}$	$90.2_{\pm 0.3}$	$\mathbf{95.3}_{\pm 0.9}$	$95.0_{\pm 0.3}$
Screw	$\mathbf{96.8}_{\pm 0.3}$	$90.9_{\pm 2.6}$	$94.1_{\pm 2.1}$	$\mathbf{96.9}_{\pm 0.3}$	$91.2_{\pm 0.8}$	$94.5_{\pm 1.0}$	$\mathbf{97.2}_{\pm 0.4}$	$92.6_{\pm 1.3}$	$94.8_{\pm 0.4}$
Tile	$99.4_{\pm 0.0}$	$\mathbf{99.8}_{\pm 0.1}$	$99.7_{\pm 0.1}$	$99.5_{\pm 0.1}$	$\mathbf{99.8}_{\pm 0.1}$	$\mathbf{99.8}_{\pm 0.0}$	$99.5_{\pm 0.1}$	$\mathbf{99.9}_{\pm 0.1}$	$99.8_{\pm 0.0}$
Toothbrush	$93.8_{\pm 0.2}$	$96.6_{\pm 4.2}$	$\mathbf{98.6}_{\pm 1.0}$	$93.5_{\pm 0.2}$	$\mathbf{99.3}_{\pm 1.3}$	$98.8_{\pm 1.3}$	$93.6_{\pm 0.0}$	$\mathbf{99.1}_{\pm 0.8}$	$98.1_{\pm 1.6}$
Transistor	$83.0_{\pm 0.3}$	$89.3_{\pm 2.9}$	$\mathbf{90.0}_{\pm 1.9}$	$83.4_{\pm 0.1}$	$90.0_{\pm 2.4}$	$\mathbf{90.6}_{\pm 1.9}$	$83.4_{\pm 0.1}$	$90.2_{\pm 2.3}$	$\mathbf{90.7}_{\pm 1.6}$
Wood	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.1}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$	$\mathbf{100}_{\pm 0.0}$
Zipper	$95.4_{\pm 0.7}$	$94.9_{\pm 6.5}$	$\mathbf{98.8}_{\pm 0.3}$	$96.1_{\pm 0.2}$	$\mathbf{99.2}_{\pm 0.2}$	$\mathbf{99.2}_{\pm 0.2}$	$96.2_{\pm 0.2}$	$99.3_{\pm 0.1}$	$\mathbf{99.4}_{\pm 0.2}$
Mean	$94.1_{\pm 0.7}$	$95.8_{\pm 1.1}$	$\mathbf{96.2}_{\pm 0.8}$	$94.6_{\pm 0.1}$	$96.8_{\pm 0.4}$	$\mathbf{97.0}_{\pm 0.2}$	$94.8_{\pm 0.1}$	$\mathbf{97.1}_{\pm 0.5}$	$97.0_{\pm 0.3}$

industrial anomaly detection. In particular, we mainly evaluate under few-shot regimes, *i.e.*, by assuming K-shot of normal images for each task. The detailed experimental setups, *e.g.*, hyperparameters, preprocessing, are provided in the Appendix A.

A. DATASETS

MVTec-AD comprises 15 sub-datasets with a total of 5,354 images, where 1,725 of which are in the test set. 15 sub-datasets are further divided into 10 object categories and 5 texture categories. VisA consists of 12 sub-datasets with 10,821 images in total. Anomalous images in VisA contain a variety of imperfections, including surface defects and structural defects. We follow the index given by Zou et al. [17] for splitting the VisA dataset into train and test sets.

B. IMPLEMENTATION DETAILS

Throughout our experiments, we use Stable Diffusion v2-1[1] as the backbone text-to-image model, which uses the CLIP text encoder for conditioning: so that compatible with our framework which utilizes CLIP as well. We use the OpenCLIP implementation[2] of CLIP ViT-H/14 model trained on the LAION-2B English subset of LAION-5B, following the choice of the Stable Diffusion v2-1 model we are based on. We use our re-implementation of WinCLIP [19] for our experiments, which we have confirmed the reproducibility of the results.

C. RESULTS

We consider a variety of existing methods as baselines for our comparison: specifically, we consider SPADE [2], PaDiM [3], PatchCore [6], and the current state-of-the-art of WinCLIP+ [19] in few-shot AD setups. For each setup, we report two versions of our method: (a) **ADP**, the default version introduced in (7) that does *not* relying on specific label texts (*e.g.*, "transistor") as considered in WinCLIP+ [19]; in addition, we also report (b) **ADP**$_\ell$ which also incorporate the knowledge of label texts. To be specific, ADP$_\ell$ utilizes the class label ℓ alongside the two learned concepts at computing importance weight (5) and constructing classification prototype vectors (6). We use *Area Under Receiver Operator Characteristic-curve* (AUROC) as the major evaluation metric. We compare the class-average AUROC on both MVTec-AD and VisA, as well as the average AUROC across the classes. We report our results with standard deviation across 3 different random seeds.

In Table 1, we report the overall performances of our methods, "ADP" and "ADP$_\ell$", for 2-shot AD compared to baselines on MVTec-AD and VisA:[3] ADP and ADP$_\ell$ significantly outperform all the baselines considered, including the state-of-the-art results of WinCLIP+ [19] on both datasets. Specifically, ADP$_\ell$ outperforms PatchCore by a margin of

[1] https://github.com/Stability-AI/stablediffusion

[2] https://github.com/openai/CLIP

[3] We report the detailed results of Table 1 in the Appendix B.

TABLE 3. Class-wise comparison of anomaly detection (AD) performance on VisA benchmark on 4-, 8-, and 16- shots. We report the mean AUROC (%) and standard deviation over three random seeds for each measurement, which highest AUROC (%) for each class is marked as bold.

	4-shot			8-shot			16-shot		
Data \ Method	WinCLIP+	**ADP**	**ADP$_\ell$**	WinCLIP+	**ADP**	**ADP$_\ell$**	WinCLIP+	**ADP**	**ADP$_\ell$**
Candle	**95.4**$_{\pm0.7}$	92.5$_{\pm1.5}$	94.0$_{\pm1.2}$	**95.6**$_{\pm0.0}$	92.8$_{\pm2.6}$	94.3$_{\pm1.9}$	**95.6**$_{\pm0.1}$	92.8$_{\pm1.3}$	94.6$_{\pm2.1}$
Capsules	81.8$_{\pm6.7}$	87.3$_{\pm0.7}$	**87.4**$_{\pm0.5}$	86.1$_{\pm0.9}$	87.1$_{\pm0.5}$	**87.7**$_{\pm0.5}$	86.9$_{\pm0.1}$	87.7$_{\pm0.8}$	**88.0**$_{\pm1.1}$
Cashew	88.9$_{\pm0.9}$	**91.7**$_{\pm1.7}$	**91.7**$_{\pm2.1}$	89.3$_{\pm0.3}$	91.4$_{\pm3.4}$	**91.6**$_{\pm3.2}$	89.3$_{\pm0.2}$	**94.9**$_{\pm0.9}$	93.9$_{\pm1.7}$
Chewinggum	95.1$_{\pm0.1}$	97.7$_{\pm0.6}$	**97.9**$_{\pm0.1}$	94.9$_{\pm0.0}$	97.6$_{\pm0.6}$	**97.8**$_{\pm0.5}$	95.0$_{\pm0.2}$	98.0$_{\pm0.9}$	**98.2**$_{\pm0.6}$
Fryum	87.7$_{\pm0.4}$	**94.6**$_{\pm2.0}$	94.0$_{\pm1.9}$	88.2$_{\pm0.6}$	**94.7**$_{\pm2.0}$	94.3$_{\pm1.7}$	88.4$_{\pm0.3}$	**94.7**$_{\pm2.1}$	94.5$_{\pm1.5}$
Macaroni1	91.3$_{\pm0.8}$	91.4$_{\pm3.3}$	**91.9**$_{\pm2.0}$	91.9$_{\pm0.1}$	92.3$_{\pm2.8}$	**92.5**$_{\pm2.1}$	92.0$_{\pm0.2}$	92.6$_{\pm3.8}$	**92.7**$_{\pm2.5}$
Macaroni2	**74.6**$_{\pm1.7}$	71.7$_{\pm3.4}$	72.5$_{\pm2.4}$	**75.6**$_{\pm0.6}$	72.9$_{\pm4.7}$	73.3$_{\pm3.3}$	**76.1**$_{\pm0.3}$	73.8$_{\pm3.2}$	73.3$_{\pm2.2}$
PCB1	88.1$_{\pm0.3}$	87.7$_{\pm1.5}$	**90.4**$_{\pm1.7}$	88.1$_{\pm0.3}$	89.6$_{\pm3.4}$	**90.9**$_{\pm2.4}$	88.5$_{\pm0.1}$	91.5$_{\pm0.6}$	**92.8**$_{\pm0.7}$
PCB2	63.1$_{\pm1.5}$	**74.3**$_{\pm2.7}$	73.8$_{\pm2.1}$	62.9$_{\pm0.3}$	**77.5**$_{\pm2.4}$	76.7$_{\pm1.8}$	63.1$_{\pm0.5}$	**80.2**$_{\pm2.6}$	79.0$_{\pm2.2}$
PCB3	70.1$_{\pm1.2}$	67.8$_{\pm9.6}$	**71.4**$_{\pm6.4}$	69.8$_{\pm0.5}$	70.5$_{\pm10.4}$	**74.1**$_{\pm7.0}$	69.5$_{\pm0.5}$	75.0$_{\pm14.0}$	**77.8**$_{\pm10.0}$
PCB4	85.6$_{\pm4.1}$	96.7$_{\pm0.8}$	**97.1**$_{\pm0.9}$	83.7$_{\pm0.3}$	97.3$_{\pm0.5}$	**97.5**$_{\pm0.6}$	82.3$_{\pm0.7}$	96.3$_{\pm1.3}$	**96.8**$_{\pm0.8}$
Pipe fryum	93.4$_{\pm0.0}$	99.1$_{\pm0.2}$	**99.2**$_{\pm0.4}$	93.6$_{\pm0.1}$	99.4$_{\pm0.2}$	**99.5**$_{\pm0.3}$	93.5$_{\pm0.1}$	99.3$_{\pm0.4}$	**99.4**$_{\pm0.3}$
Mean	84.6$_{\pm0.4}$	87.7$_{\pm0.3}$	**88.4**$_{\pm0.4}$	85.0$_{\pm0.0}$	88.6$_{\pm0.3}$	**89.2**$_{\pm0.1}$	85.0$_{\pm0.1}$	89.7$_{\pm0.9}$	**90.1**$_{\pm0.5}$

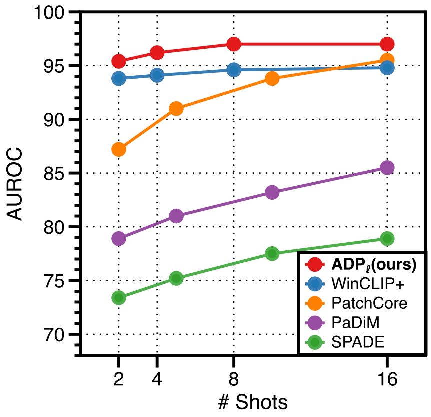

FIGURE 2. Comparison of AUROC (%) on MVTec-AD benchmark. The results of SPADE, PaDiM and PatchCore are from those reported by Roth et al. [6], *viz.*, on 2-,5-,10- and 16- shots.

9.1% on MVTec-AD and 5.3% on VisA in AUROC, which has been the state-of-the-art in full-shot anomaly detection and that was a state-of-the-art approach even in few-shot AD before WinCLIP+.

Table 2 and 3 further compare the methods in 4-, 8-, and 16-shot setups of MVTec-AD and VisA, respectively, and Fig. 2 compares the performance trends in plots. Here, in the tables we report class-wise AUROC across all the object classes of the benchmarks.[4] Along a similar trend with the 2-shot results, ADP and ADP$_\ell$ could still significantly and consistently outperform the state-of-the-art results of WinCLIP+ in all the setups considered. On both MVTec-AD and VisA, we observe that our approach of ADP exhibits a wider performance gap over WinCLIP+ as more shots are given: specifically, on the 4-shot MVTec-AD, ADP outperforms WinCLIP+ by 1.7% in AUROC, while it does by 2.3% on the 16-shot scenario. Similarly, in the case of VisA, ADP improves over WinCLIP+ by 3.1% in AUROC on 4-shot, while it does by 4.7% on the 16-shot setup. Regarding the performance of ADP$_\ell$ over ADP: although the knowledge of label texts in ADP$_\ell$ does helpful to improve our results on low-shot setups, *e.g.*, 4-shot, we observe that ADP gradually matches the performance with ADP$_\ell$ having with more shots: in the 16-shot scenario of MVTec-AD, ADP even shows a consistently better performances over ADP$_\ell$ when viewed in class-wise, achieving 97.1% in AUROC.

TABLE 4. Comparison of AUROC (%) with naïve textual inversion on 4-shot. Naïve textual inversion is denoted as "TI".

Data \ Method	TI	ADP	ADP$_\ell$
MVTec-AD	88.6	96.0	**96.6**
VisA	78.5	87.7	**88.7**

TABLE 5. Comparison of AUROC (%) across the use of learned concepts and labels for 2- and 8-shot.

$\mathbf{c}_n$	$\mathbf{c}_a$	label	$K=2$	$K=8$
✓	✗	✗	94.1	96.6
✗	✓	✗	94.2	95.5
✓	✓	✗	94.7	**97.1**
✓	✓	✓	**95.7**	**97.1**

D. ABLATION STUDY

1) COMPARISON WITH TEXTUAL INVERSION

In Table 4, we compare our proposed ADP with the standard textual inversion [21] in the context of AD. Specifically, we compare our results on 4-shot MVTec-AD and VisA with an ablation that the steps for concept optimization are replaced by the standard version of textual inversion (as reported by "TI" in Table 4). Overall, we observe that converting only via textual inversion, *e.g.*, encoding tokens simply through the reconstruction loss, falls short specifically in few-shot AD. For example, on the VisA dataset we observe that ADP improves upon the original textual inversion by 9.2% in AUROC. These results highlight the suitability of ADP to effectively capture the concepts related to *abnormality* into tokens through an additional guidance via CLIP.

[4] We report the detailed results of Table 2 and Table 3 in the Appendix B.

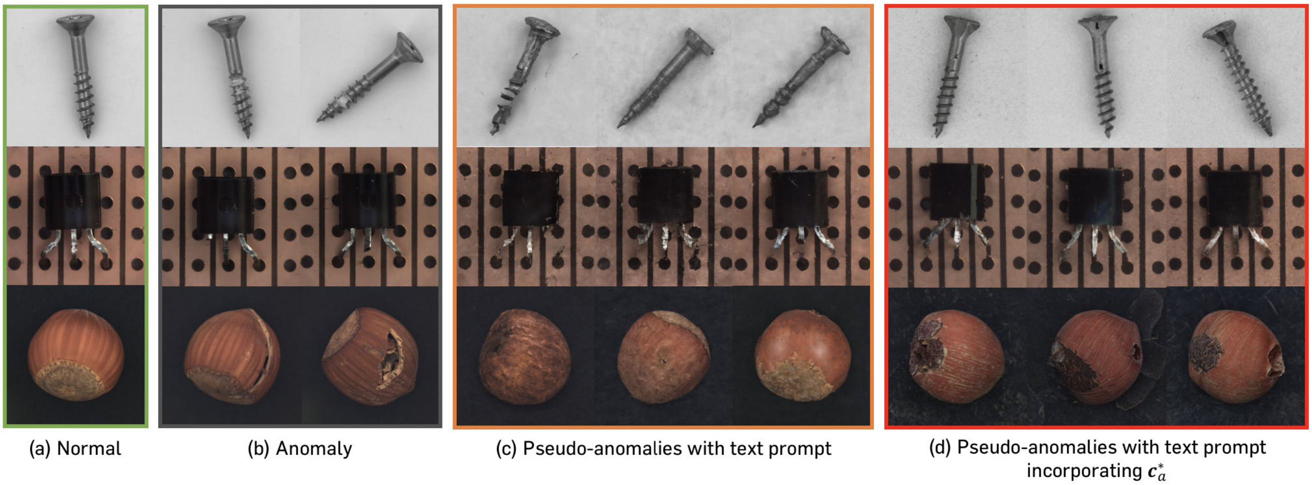
(a) Normal
(b) Anomaly
(c) Pseudo-anomalies with text prompt
(d) Pseudo-anomalies with text prompt incorporating c_a^*

FIGURE 3. Visualization of (a) normal, (b) anomaly and (c-d) pseudo anomalies synthesized via text-guided image manipulation [34]. (c) is conditioned with simple text prompt, *e.g.*, "a photo with damage", and (d) is conditioned with prompt incorporating concept $\mathbf{c}_a^*$, *e.g.*, "a photo of a $\mathbf{c}_a^*$ with damage". (d) shows that incorporating $\mathbf{c}_a^*$ in the conditioning prompt leads to produce fine-grained anomalies closely resemble real anomalies, while (c) shows collapse through the entire image.

2) EFFECT OF CONCEPT LEARNING

We also conduct a detailed ablation study to further assess the effectiveness of our learned concepts, specifically, those from (a) the *normal state* learning (Section IV-A) and (b) the *anomaly state aware* learning (Section IV-B) schemes. Specifically, we evaluate on MVTec-AD with the 2- and 8-shot setups, to compare the behaviors on both lower- and higher-shot regimes in applying our method. The results are summarized in Table 5. In the first and the second rows of the table, we examine the results obtained by incorporating only $\mathbf{c}_n$ and $\mathbf{c}_a$ in the text prompts for AD. The results on the third and last row indicate ours, *viz.*, which correspond to ADP and ADP$_\ell$, respectively. Overall, the results shows that mixing $\mathbf{c}_n$ and $\mathbf{c}_a$ via ADP (Section IV-C) leads to a better performance, confirming the effectiveness of our self-tuning scheme. We observe that the performance itself of $\mathbf{c}_a$ as an individual concept may not be significantly better compared to $\mathbf{c}_n$. A clear performance gain could be obtained by combining the two concepts, however, confirming that $\mathbf{c}_n$ and $\mathbf{c}_a$ complement each other. With extremely low shots of samples, *e.g.*, $K = 2$, utilizing the label texts could further improves performance. In the case of $K = 8$, however, such a gain diminishes and using only the two learned tokens, $\mathbf{c}_n$ and $\mathbf{c}_a$, could already yield comparable performances. This observation supports our initial hypothesis on the limitation of naïve language-based approaches for larger-shot AD, and the effectiveness of our method upon this.

3) SYNTHETIC PSEUDO-ANOMALIES

Prior studies have proposed synthesizing *anomalous* images by adding visually irregular appearances into normal images [4], [7], [35]. In this paper, we take a different approach which generates pseudo-anomalies using a pre-trained text-to-image diffusion model [34]. Specifically, this is achieved by adding noise to a given reference image and conditioning the reconstruction process on text prompts. To evaluate the efficacy of synthetic pseudo-anomalies, we conduct an ablation study integrating real anomaly images into the process of learning the anomaly-aware concept, $\mathbf{c}_a^*$. The results are presented in Table 6. As expected, the use of real anomaly images further enhance performance, while pseudo-anomalies bring similar effects to obtaining the anomaly-aware concept, resulting in a 0.6% improvement compared to solely utilizing normal images.

Furthermore, to qualitatively assess the usage of anomaly-aware concept $\mathbf{c}_a^*$ in generating pseudo-anomalies, we investigate two types of prompts: (1) simple text prompts, such as "a photo with damage", (2) prompts incorporating the *anomaly-aware* concept $\mathbf{c}_a^*$, such as "a photo of a $\mathbf{c}_a^*$ with damage". The results presented in Fig. 3 demonstrate that the use of *anomaly-aware* concept $\mathbf{c}_a^*$ leads to the generation of fine-grained anomalies, compared to simple text prompts. For example, anomalous "hazelnut" generated with prompts containing $\mathbf{c}_a$, exhibit small scars or holes while generated images with simple text prompt exhibits coarse-grained transformation in images. Furthermore, generated anomalies from the simple text prompts exhibit unexpected defects not only in the object but also in the background, while the use of $\mathbf{c}_a$ guides the model to produce anomalies that closely resemble real anomalies, focusing primarily on defects within objects.

4) PROMPT TEMPLATE SELECTION

We primarily use the same prompt templates proposed in prior works, *viz.*, Textual Inversion [21]. To further verify the robustness of ADP regarding to the selection of prompt templates, we conduct an ablation study with different set of prompt templates. Specifically, we randomly choose a different number of templates from the entire set (detailed in Appendix A) and also conduct an additional experiment only

TABLE 6. Comparison of AUROC (%) across the usage of generated pseudo-anomalies and real anomaly images for obtaining anomaly-aware concept $\mathbf{c}_a^*$ on 2-shot.

	Pseudo-anomalies		Real anomalies	
Data \ Method	✗	✓	2	4
MVTec-AD	94.1	94.7	94.3	94.9

TABLE 7. Comparison of AUROC (%) across the selection of prompt templates on 4-shot. ADP-T incorporates T number of templates where ADP-27 is the default setting for our main experiment.

Data \ Method	ADP-27	ADP-9	ADP-3	ADP-1
MVTec-AD	96.0	95.6	95.6	95.2

using a single prompt template ("a photo of a { }"). The results are reported in Table 7, which confirms that the choice of prompt templates brings marginal effects for the performance of ADP.

VI. CONCLUSION

In this paper, we propose *Anomaly Detection via Personalization* (ADP), a novel approach to address the challenging problem of few-shot industrial anomaly detection based on recent text-to-image diffusion models. We show that aligning state prompts with image features effectively guides the model to learn concepts related to *normal* and *anomalous* instances. Additionally, we introduce synthesizing pseudo-anomalies using a personalized generative model based on the learned concepts. By incorporating these pseudo-anomalies, ADP further optimizes the use of concepts with simple self-tuning scheme. ADP could outperform state-of-the-arts in recent few-shot benchmarks. Moreover, ADP can be applied in scenarios where text labels are scarce, without experiencing a significant drop compared to using the label. We believe our work could shed a light in exploring model personalization for downstream tasks beyond generative modeling.

Limitation: Despite its strong performances in few-shot AD, we expect the effectiveness of current ADP may saturate earlier as more normal samples become available: the current technique of textual inversion is known to fall short with many samples, *e.g.*, more than 4-5 in practice [21]. Making textual inversion to extract better concepts from many samples would be an interesting future work itself, not only in the context of AD but also in the context of generative modeling.

APPENDIX A
IMPLEMENTATION DETAILS

A. CONCEPT LEARNING

Unless specified otherwise, we maintain the original hyperparameter choices from LDM [24]. The batch size is set to 4, the base learning rate is set to 5.0×10^{-4} and all results are obtained after 3,000 optimization steps. Both MVTec-AD [16] and VisA [17] datasets are resized to a resolution of 512 × 512.

For concept learning, hyperparameters α and γ are consistent through the entire experiment. For normal-aware concept learning, α, regularization hyperparameter for aligning state prompts with images, is set to 0.003, which shows similar scale to reconstruction loss. We find that applying large α value results over-fitting to normal state prompt, *e.g.*, "a photo of a flawless $\mathbf{c}_n^*$", with the given image. For anomaly-aware concept learning, we first synthesize pseudo-anomalies via pre-trained text-to-image diffusion model [34]. Specifically, with the given reference images, we set the *strength* parameter, *i.e.*, the amount of noise initially added to the given image, as 0.5. The guidance scale and number of inference steps are set to 7.5 and 30 respectively. We explore diverse amount of noise, and set which is distinguishable with normal samples while maintaining the high-level features of the images. In Fig. 5 and Fig. 7, pseudo-anomalies with different strength is shown. Hyperparameter α is set to 0.002 and γ, which serves as margin of the CLIP-based repel loss, is set and 0.8.

TABLE 8. Model size and mean inference time per image on MVTec-AD. For WinCLIP and ADP, experiments are conducted under 4-shot setting.

Method	PaDiM	PatchCore	WinCLIP+	ADP
Model size (MB) Time (s)	263.03 0.57	263.03 0.18	3761.71 0.81	3761.71 0.73

B. ANOMALY DETECTION WITH LEARNED CONCEPTS

For self-tuning, we generate 20 pseudo-anomalies for pseudo-validation set. We set *strength* parameter, *i.e.*, the amount of noise initially added to the given image, as 0.5. The guidance scale and number of inference steps are set to 7.5 and 30 respectively. We also explore the impact of the size of pseudo-validation set, *i.e.*, number of generated pseudo-anomalies in the pseudo-validation set.[5] ADP demonstrates consistent performance regardless of the number of generated pseudo-anomalies in the pseudo-validation set.

We employ the data pre-processing pipeline from OpenCLIP [36] for both MVTec-AD and VisA datasets. This pipeline includes channel-wise standardization using the pre-computed mean `[0.48145466, 0.4578275, 0.40821073]` and standard deviation `[0.26862954, 0.26130258, 0.27577711]` after normalizing each RGB image to the range of [0, 1]. Additionally, we set the input resolution to be 224 by default, regardless of the original size of the input image. When reproducing the results for WinCLIP+ [19], we follow the same pre-processing pipeline to ensure compatibility in our experiments.

C. COMPUTATION

1) TRAINING

We use 64 CPU cores (Intel Xeon CPU @ 2.90GHz) and 1 GPU (NVIDIA GeForce RTX 3090 24GB GPU) for performing concept learning. The training for 3,000 optimization steps takes approximately 1.5 hours for each class. We need two times of concept learning *i.e.*, normal-aware

[5] ADP consistently achieves 96.0% in AUROC under varying number of psuedo-anomlies (n = 5, 10, 20 and 40) on MVTec-AD 4-shot setting.

TABLE 9. Comparison of anomaly detection (AD) in terms of class-wise AUROC (%) on MVTec-AD for 2- and 4-shot.

	2-shot						4-shot					
Data \ Method	SPADE	PaDiM	PatchCore	WinCLIP+	**ADP**	**ADP$_\ell$**	SPADE	PaDiM	PatchCore	WinCLIP+	**ADP**	**ADP$_\ell$**
Bottle	99.5±0.1	98.5±1.0	99.2±0.3	93.3±0.1	97.1±1.5	95.1±0.4	99.5±0.2	98.8±0.2	99.2±0.3	93.4±0.3	98.9±0.4	97.2±0.7
Cable	76.2±5.2	62.3±5.9	91.0±2.7	82.6±0.2	85.7±1.5	86.2±1.8	83.4±3.1	70.0±6.1	91.0±2.7	83.0±0.0	87.9±2.5	88.3±3.1
Capsule	70.9±6.1	64.3±3.0	72.8±7.0	84.2±9.0	85.0±12.9	85.3±12.1	78.9±5.5	65.2±2.5	72.8±7.0	84.4±9.4	83.4±11.9	84.0±11.5
Carpet	98.3±0.4	97.8±0.5	96.6±0.5	100±0.0	100±0.0	100±0.0	98.6±0.2	97.9±0.4	96.6±0.5	100±0.0	99.9±0.1	100±0.0
Grid	41.3±3.6	67.2±4.2	67.7±8.3	99.2±0.0	97.4±0.7	98.6±0.0	44.6±6.6	68.1±3.8	67.7±8.3	99.1±0.2	98.0±2.5	99.5±0.6
Hazelnut	96.2±2.1	90.8±0.8	93.2±3.8	97.0±0.6	98.8±0.9	98.3±0.9	98.4±1.3	91.9±1.2	93.2±3.8	97.5±0.1	99.4±0.5	98.9±0.4
Leather	100±0.0	97.5±0.9	97.9±0.7	100±0.0	93.1±11.9	100±0.0	100±0.0	98.5±0.2	97.9±0.7	100±0.0	100±0.0	100±0.0
Metal nut	77.0±7.9	54.8±3.8	77.7±8.5	95.5±0.3	99.7±0.3	99.1±0.1	77.8±5.7	60.7±5.2	77.7±8.5	95.7±0.3	99.4±0.5	99.6±0.2
Pill	84.8±0.9	59.1±6.4	82.9±2.9	90.0±0.2	95.2±0.4	95.2±1.0	86.7±0.3	54.9±2.7	82.9±2.9	90.1±0.1	95.2±0.3	94.9±0.6
Screw	46.6±2.2	54.0±4.4	49.0±3.8	96.5±0.2	91.9±5.6	94.8±3.3	50.5±5.4	50.0±4.1	49.0±3.8	96.8±0.3	90.9±2.6	94.1±2.1
Tile	99.9±0.1	93.3±1.1	98.5±1.0	99.4±0.0	99.5±0.2	99.6±0.1	100±0.0	93.1±0.6	98.5±1.0	99.4±0.0	99.8±0.1	99.7±0.1
Toothbrush	78.6±3.2	87.6±4.2	85.9±3.5	94.0±0.6	88.5±3.8	95.0±4.3	78.8±5.2	89.2±2.5	85.9±3.5	93.8±0.2	96.6±4.2	98.6±1.0
Transistor	83.4±3.8	81.3±3.7	72.8±6.3	82.4±0.4	82.3±5.7	87.5±2.4	81.4±2.1	82.4±6.5	90.0±4.3	83.0±0.3	89.3±2.9	90.0±1.9
Wood	99.2±0.4	96.9±0.5	98.3±0.6	100±0.0	99.9±0.3	99.9±0.1	98.9±0.6	97.0±0.2	98.3±0.6	100±0.0	100±0.0	100±0.0
Zipper	93.3±2.9	86.3±2.6	94.0±2.1	92.4±4.4	95.5±5.6	95.8±5.4	95.1±1.3	88.3±2.0	94.0±2.1	95.4±0.7	94.9±6.5	98.8±0.3
Mean	82.9±2.6	78.9±3.1	86.3±3.3	93.8±1.0	94.4±1.2	95.4±0.9	84.8±2.5	80.4±2.5	88.8±2.6	94.1±0.7	95.8±1.1	96.2±0.8

TABLE 10. Comparison of anomaly detection (AD) in terms of class-wise AUROC (%) on VisA for 2- and 4-shot.

	2-shot						4-shot					
Data \ Method	SPADE	PaDiM	PatchCore	WinCLIP+	**ADP**	**ADP$_\ell$**	SPADE	PaDiM	PatchCore	WinCLIP+	**ADP**	**ADP$_\ell$**
Candle	91.3±3.3	75.8±2.1	85.3±1.5	95.3±0.4	94.1±1.6	95.1±1.6	92.8±2.1	77.5±1.6	87.8±0.8	95.4±0.7	92.5±1.5	94.0±1.2
Capsules	71.7±11.2	51.7±4.6	57.8±5.4	82.2±5.5	84.4±4.1	84.7±4.1	73.4±7.1	52.7±3.4	63.4±5.4	81.8±6.7	87.3±0.7	87.4±0.5
Cashew	97.3±1.4	74.6±3.6	93.6±0.6	88.9±0.8	91.5±4.3	91.6±3.9	96.4±1.3	77.7±3.2	93.0±1.5	88.9±0.9	91.7±1.7	91.7±2.1
Chewinggum	93.4±1.0	82.7±2.1	97.8±0.6	94.6±0.3	98.2±0.5	98.1±0.7	93.5±1.4	83.5±3.7	98.3±0.3	95.1±0.1	97.7±0.6	97.9±0.1
Fryum	90.5±3.9	69.2±9.0	83.4±2.4	87.7±0.3	93.6±1.4	91.9±2.1	92.9±1.6	71.2±5.9	88.6±1.3	87.7±0.4	94.6±2.0	94.0±1.9
Macaroni1	69.1±8.2	62.2±5.0	75.6±4.6	91.1±0.6	91.1±3.7	92.9±3.4	65.8±1.2	65.9±3.9	82.9±2.7	91.3±0.8	91.4±3.3	91.9±2.0
Macaroni2	58.3±4.4	50.8±2.9	57.3±5.6	74.7±1.5	76.1±4.7	76.7±5.2	56.7±3.2	55.0±2.9	61.7±1.8	74.6±1.7	71.7±3.4	72.5±2.4
PCB1	86.7±1.1	62.4±10.8	71.5±20.0	87.7±0.4	80.1±13.4	83.9±9.4	83.4±8.5	82.6±1.5	84.7±6.7	88.1±0.3	87.7±1.5	90.4±1.7
PCB2	70.3±8.1	66.8±2.0	84.3±1.7	61.9±1.6	71.3±3.0	71.1±2.9	71.7±7.0	73.5±2.4	84.3±1.0	63.1±1.5	74.3±2.7	73.8±2.1
PCB3	75.8±5.7	67.3±3.8	84.8±1.2	70.2±0.5	64.0±1.0	67.0±2.6	79.0±4.1	65.9±1.9	87.0±1.1	70.1±1.2	67.8±9.6	71.4±6.4
PCB4	86.1±8.2	69.3±13.7	94.3±3.2	83.0±5.2	86.3±10.6	90.4±6.2	95.4±2.3	85.4±2.0	95.6±1.6	85.6±4.1	96.7±0.8	97.1±0.9
Pipe fryum	78.1±3.0	75.3±1.8	93.5±1.3	93.3±0.1	98.3±1.9	98.9±1.1	79.3±0.9	82.9±2.2	96.4±0.7	93.4±0.0	99.1±0.2	99.2±0.4
Mean	80.7±5.0	67.4±5.1	81.6±4.0	84.2±0.2	85.7±0.9	86.9±0.9	81.7±3.4	72.8±2.9	85.3±2.1	84.6±0.4	87.7±0.3	88.4±0.4

TABLE 11. Comparison with existing many-shot AD methods in terms of AUROC (%) on MVTec-AD.

	8-shot					
Data \ Method	TDG+	DiffNet+	RegAD	WinCLIP+	**ADP**	**ADP$_\ell$**
Bottle	70.3	99.4	99.8	93.7±0.1	99.4±0.3	97.5±1.0
Cable	74.7	87.9	80.6	83.0±0.1	88.0±1.9	88.5±2.4
Capsule	44.7	78.6	76.3	90.9±1.4	93.1±1.7	93.0±1.5
Carpet	78.2	78.5	98.5	100±0.0	99.5±0.7	99.7±0.4
Grid	87.6	78.5	91.5	99.0±0.5	98.2±1.7	99.4±0.4
Hazelnut	82.8	97.9	96.5	97.7±0.1	99.5±0.7	99.1±0.5
Leather	93.5	92.2	100	100±0.0	100±0.0	100±0.0
Metal nut	68.7	67.6	98.3	95.8±0.4	99.6±0.4	99.6±0.2
Pill	67.9	82.1	80.6	90.1±0.1	95.6±0.5	94.9±0.5
Screw	99.0	75.0	63.4	96.9±0.3	91.2±0.8	94.5±1.0
Tile	87.4	99.6	97.4	99.5±0.1	99.8±0.1	99.8±0.0
Toothbrush	57.6	60.8	98.5	93.5±0.2	99.3±1.3	98.8±1.3
Transistor	71.5	63.3	93.4	83.4±0.1	90.0±2.4	90.6±1.9
Wood	98.4	99.4	99.4	100±0.0	100±0.1	100±0.0
Zipper	66.3	87.3	94.0	96.1±0.2	99.2±0.2	99.2±0.2
Mean	76.6	83.2	91.2	94.6±0.1	96.8±0.4	97.0±0.2

concept learning and anomaly-aware concept learning, which takes similar time for each concept learning. For self-tuning, the computation of a single image takes around 0.7 seconds, with each class containing 20 pseudo-anomalies and 2 to 16 normal images, depending on the experimental setting.

2) INFERENCE

We report the inference time of ADP, an essential part in industrial application. The results are presented in Table 8 comparing the re-implementations of PaDiM [3], PatchCore [6] using WideResNet50 and WinCLIP [19] using CLIP ViT-H/14. ADP utilizes CLIP ViT-H/14 model for

TABLE 12. Comparison with existing many-shot AD methods in terms of AUROC (%) on MVTec-AD.

Methods	Setup	AD
ADP (ours)	2-shot	94.4
ADP (ours)	4-shot	95.8
ADP (ours)	8-shot	96.8
ADP (ours)	16-shot	97.1
RegAD	8shot	91.2
GraphCore	8shot	95.9
TDG+	16-shot	78.0
DiffNet+	16-shot	87.3
MKD	full-shot	87.7
P-SVDD	full-shot	92.1
CutPaste	full-shot	95.2
Metaformer	full-shot	95.8
PatchCore	full-shot	99.6

TABLE 13. Comparison of anomaly detection (AD) in terms of class-wise AUROC (%) with naïve textual inversion and across the use of learned concepts in MVTec-AD for 4-shot. Naïve textual inversion is denoted as "TI".

Data \ Method	TI	$\mathbf{c}_n^*$	$\mathbf{c}_a^*$	$\mathbf{c}_n^* + \mathbf{c}_a^*$
Bottle	91.5	99.4	97.9	98.6
Cable	76.3	90.3	90.9	90.6
Capsule	61.8	88.5	90.0	88.6
Carpet	100	100	100	100
Grid	99.6	99.0	93.7	95.2
Hazelnut	94.4	99.7	98.4	99.8
Leather	88.0	100	100	100
Metal nut	98.8	98.0	99.2	98.9
Pill	84.5	94.4	95.4	95.1
Screw	92.1	84.6	86.2	88.8
Tile	99.4	99.5	99.6	99.8
Toothbrush	75.0	99.4	92.8	100
Transistor	71.3	86.9	86.0	86.3
Wood	98.1	100	100	100
Zipper	98.5	98.5	98.3	98.4
Mean	88.6	95.9	95.2	96.0

TABLE 14. Comparison of anomaly detection (AD) in terms of class-wise AUROC (%) with naïve textual inversion and across the use of learned concepts in VisA for 4-shot. Naïve textual inversion is denoted as "TI".

Data \ Method	TI	$\mathbf{c}_n^*$	$\mathbf{c}_a^*$	$\mathbf{c}_n^* + \mathbf{c}_a^*$
Candle	97.0	90.0	91.4	90.9
Capsules	88.0	88.5	77.6	87.5
Cashew	76.6	92.4	91.5	92.6
Chewinggum	97.4	98.8	96.9	97.6
Fryum	51.1	96.3	96.1	96.4
Macaroni1	84.9	92.9	81.7	87.7
Macaroni2	66.4	62.0	69.0	67.9
PCB1	60.5	85.0	91.4	88.8
PCB2	65.9	77.6	63.6	72.9
PCB3	68.0	69.4	69.9	75.0
PCB4	88.9	94.3	96.1	96.3
Pipe fryum	97.6	98.1	99.0	99.0
Mean	78.5	87.1	85.3	87.7

anomaly detection, which is same as the one used by WinCLIP. As can be seen, inference time of ADP does not require significant overhead considering the model size compared to other AD methods.

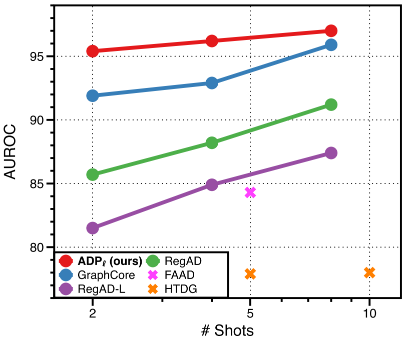

FIGURE 4. Comparison of AUROC(%) on MVTec-AD benchmark with existing few-shot AD methods.

D. PROMPT TEMPLATES

Below we provide the list of text templates used when learning the state-aware concept and detecting anomaly where $S \in \{S_n, S_c\}$ are state templates and $\mathbf{c} \in \{\mathbf{c}_n^*, \mathbf{c}_a^*\}$ are concepts:

- "a photo of a $S(\mathbf{c})$",
- "a rendering of a $S(\mathbf{c})$.",
- "a cropped photo of the $S(\mathbf{c})$.",
- "the photo of a $S(\mathbf{c})$.",
- "a photo of a clean $S(\mathbf{c})$.",
- "a photo of a dirty $S(\mathbf{c})$.",
- "a dark photo of the $S(\mathbf{c})$.",
- "a photo of my $S(\mathbf{c})$.",
- "a photo of the cool $S(\mathbf{c})$.",
- "a close-up photo of a $S(\mathbf{c})$.",
- "a bright photo of the $S(\mathbf{c})$.",
- "a cropped photo of a $S(\mathbf{c})$.",
- "a photo of the $S(\mathbf{c})$.",
- "a good photo of the $S(\mathbf{c})$.",
- "a photo of one $S(\mathbf{c})$.",
- "a close-up photo of the $S(\mathbf{c})$.",
- "a rendition of the $S(\mathbf{c})$.",
- "a photo of the clean $S(\mathbf{c})$.",
- "a rendition of a $S(\mathbf{c})$.",
- "a photo of a nice $S(\mathbf{c})$.",
- "a good photo of a $S(\mathbf{c})$.",
- "a photo of the nice $S(\mathbf{c})$.",
- "a photo of the small $S(\mathbf{c})$.",
- "a photo of the weird $S(\mathbf{c})$.",
- "a photo of the large $S(\mathbf{c})$.",
- "a photo of a cool $S(\mathbf{c})$.",
- "a photo of a small $S(\mathbf{c})$.",

APPENDIX B ADDITIONAL RESULTS

A. QUANTITATIVE RESULTS

1) CLASS-WISE COMPARISON

We provide a detailed anomaly detection (AD) performance, specifically in terms of class-wise AUROC (%). For the 2-shot and 4-shot scenarios, we report the mean and standard deviation over three random seeds for WinCLIP+ [19], ADP and ADP$_\ell$, while other baselines (SPADE [2], PaDiM [3] and

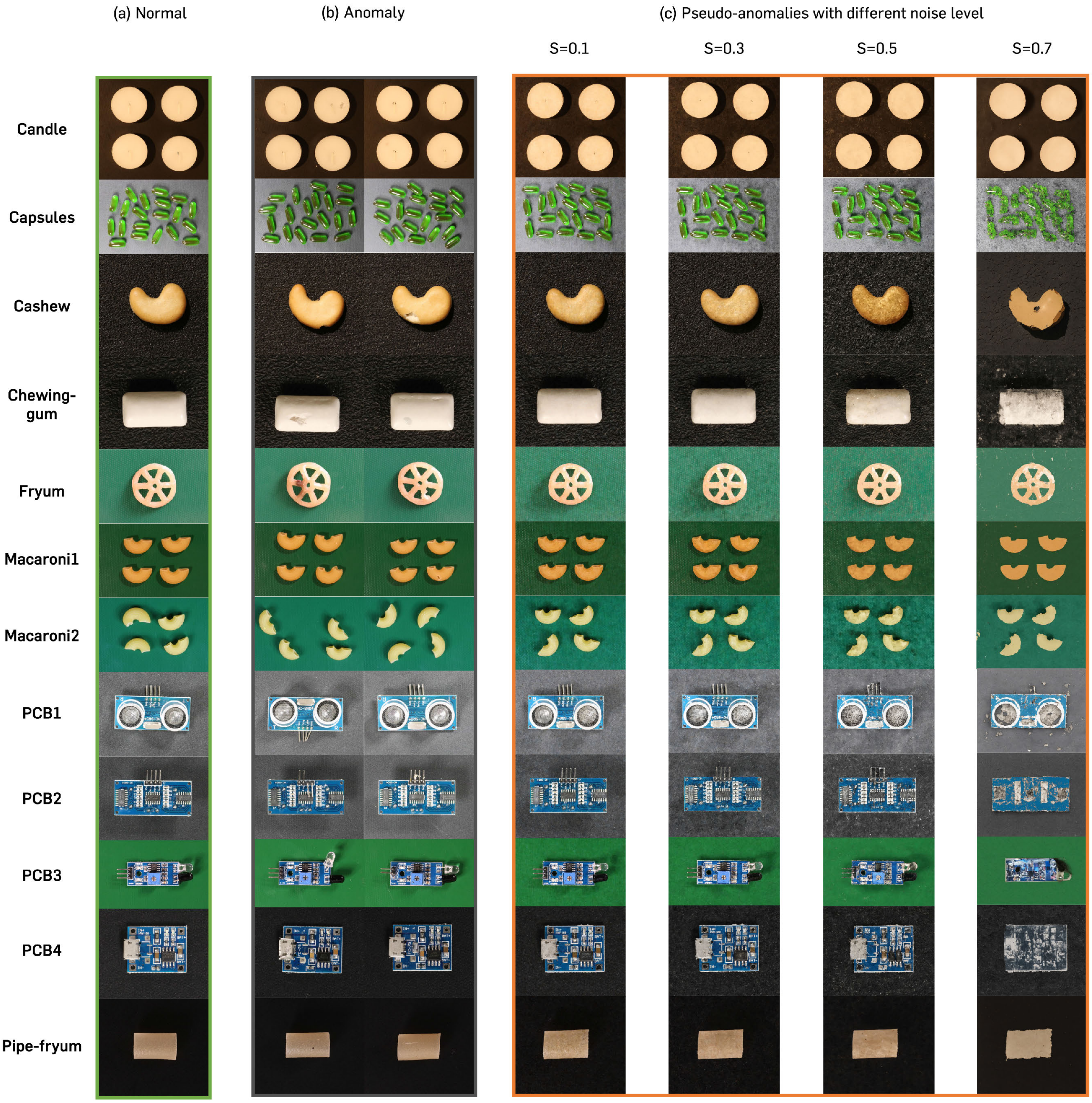

FIGURE 5. Visualization of (a) normal, (b) anomaly and (c) pseudo-anomalies synthesized via pre-trained text-to-image diffusion model with different noise level (S) in VisA. Pseudo-anomalies are generated with simple prompt text such as "a photo with damage".

PatchCore [6]) are from those reported by Jeong et al. [19]. The class-wise AUROC (%) results for the MVTec-AD dataset are presented in Table 9 for the 2-shot and 4-shot settings. Similarly, the class-wise AUROC results for the VisA dataset can be found in Table 10 for the 2-shot and 4-shot settings. Additionally, we compare our 8-shot AD results with other 8-shot AD methods on MVTec-AD dataset in Table 11. The results for TDG+ [8], DiffNet+ [26] and RegAD [27] are from the work of Huang et al. [27].

2) ADDITIONAL COMPARISON WITH EXISTING AD METHODS

Fig. 4 presents the additional comparison including recent few-shot AD methods [8], [27], [37], [38]. As can be seen, ADP$_\ell$ consistently exhibits superior performance, particularly in extreme few-shot regime, such as the 2-shot setting. Table 12 provides a extensive comparison including the full-shot results of various prior works on the MVTec-AD dataset. In the 4-shot scenario, ADP surpasses the performance of CutPaste [4], a recent full-shot method for AD and is

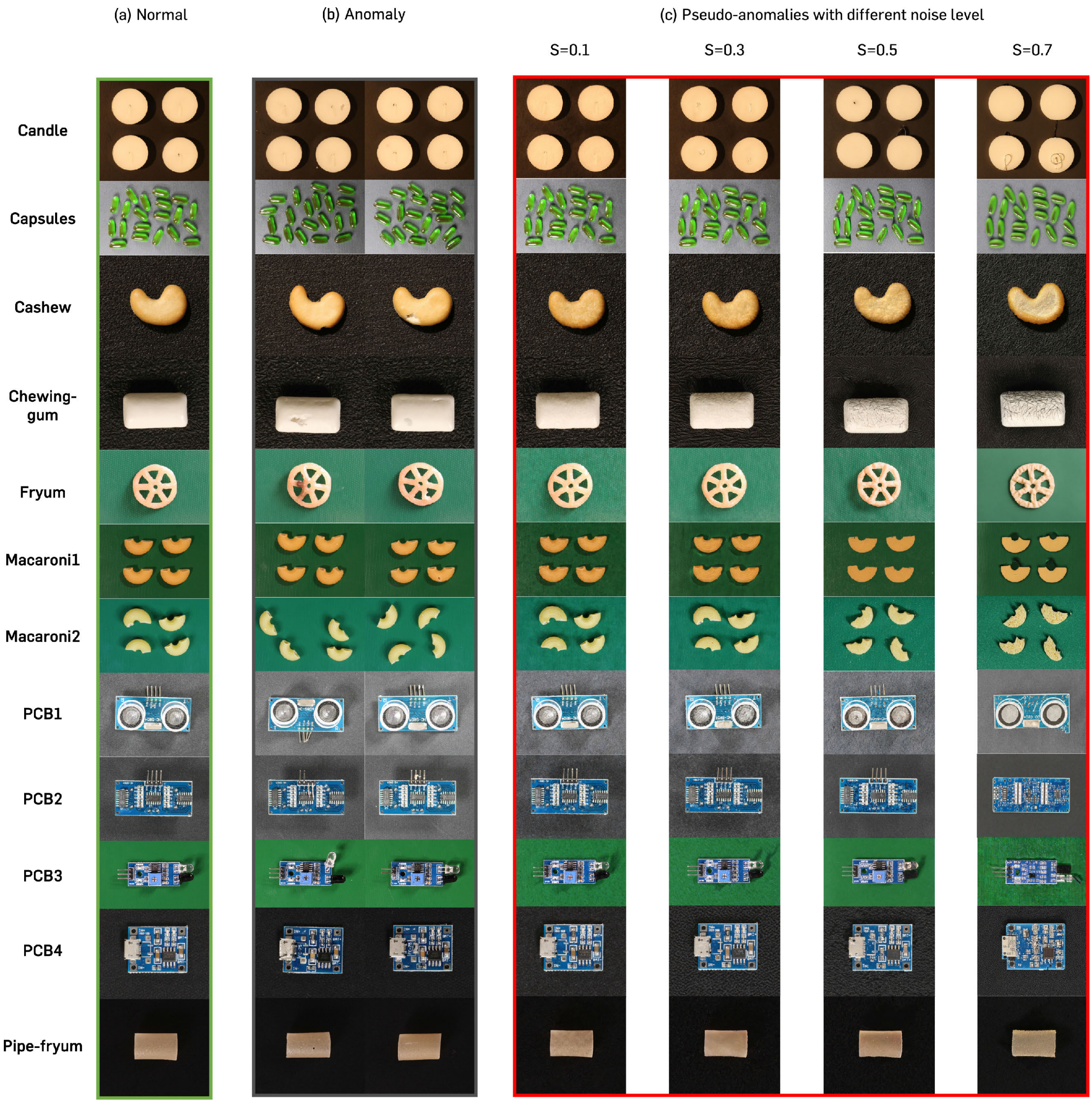

FIGURE 6. **Visualization of (a) normal, (b) anomaly and (c) pseudo-anomalies synthesized via pre-trained text-to-image diffusion model with different noise level (S) in VisA. Pseudo-anomalies are generated with prompts incorporating $\mathbf{c}_a^*$, such as "a photo of a $\mathbf{c}_a^*$ with damage".**

competitive with Metaformer [39]. Furthermore, our 4-shot ADP achieves superior performance compared to recent few-shot AD methods incorporating 16 shots such as TDG+, DiffNet+, and RegAD.

3) COMPARISON WITH TEXTUAL INVERSION

Table 13 and 14 present a class-wise comparison between standard textual inversion (referred to as "TI" in Table 13 and 14) and the utilization of learned concepts. The evaluation is conducted on 4-shot anomaly detection tasks in MVTec-AD and VisA datasets, respectively. The results are represented by $\mathbf{c}_n^*$ and $\mathbf{c}_a^*$, which indicate the outcomes obtained by incorporating only $\mathbf{c}_n^*$ and $\mathbf{c}_a^*$ in the text prompts. Furthermore, $\mathbf{c}_n^* + \mathbf{c}_a^*$ represents the combination of both concepts via ADP (Section IV-C). In general, the inclusion of concepts leads to a notable improvement in anomaly detection performance. While the utilization of only $\mathbf{c}_a^*$ does not yield significant enhancements, combining $\mathbf{c}_n^*$ and $\mathbf{c}_a^*$ proves to be mutually beneficial. Specifically, the incorporation of learned concepts proves effective in identifying fine-grained anomalies, such as the "Capsule" class in the MVTec-AD dataset or the "PCB" classes in the VisA dataset.

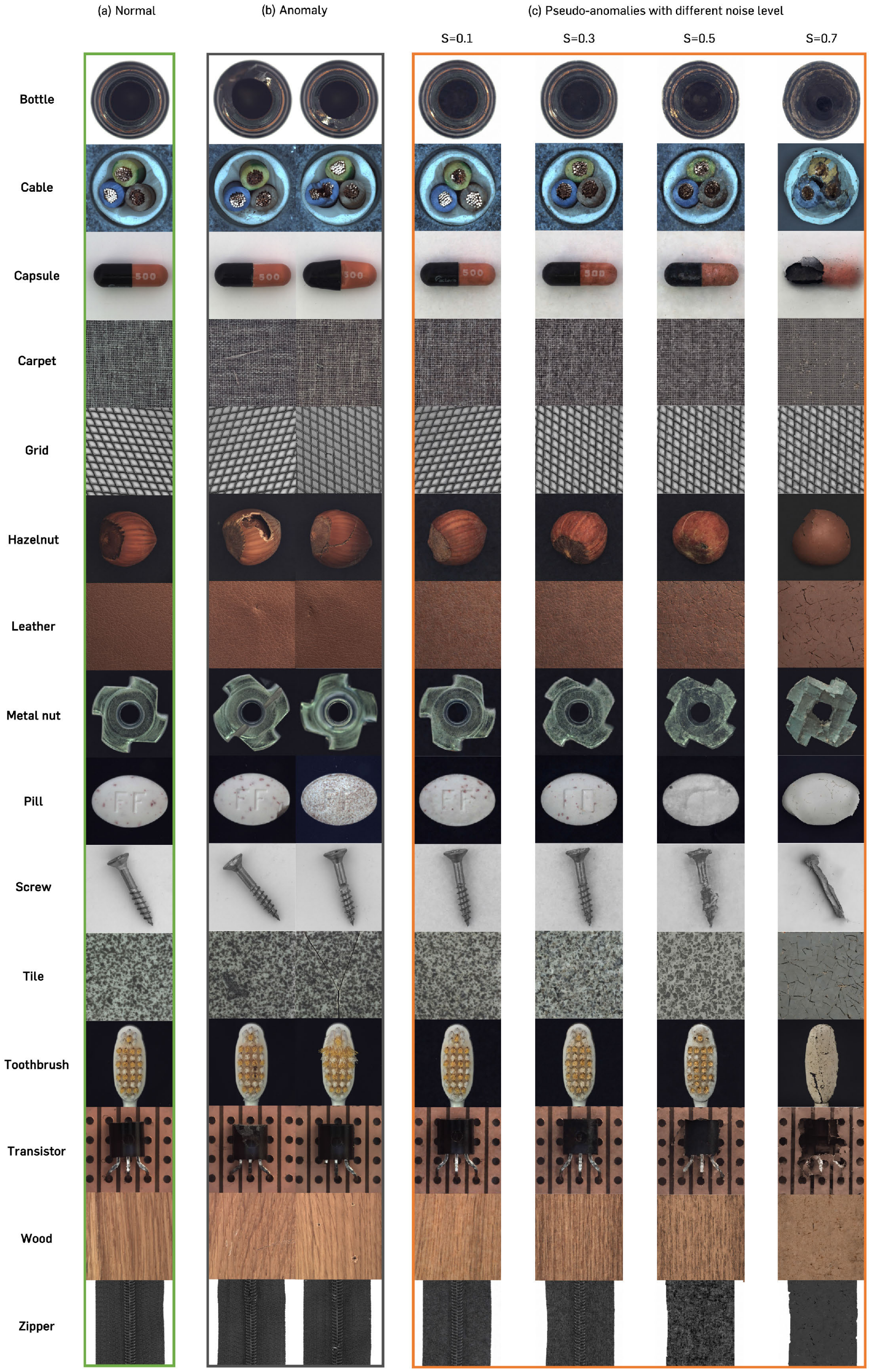

FIGURE 7. Visualization of (a) normal, (b) anomaly and (c) pseudo-anomalies synthesized via pre-trained text-to-image diffusion model with different noise level (S) in MVTec-AD. Pseudo-anomalies are generated with **simple prompt text** such as **"a photo with damage"**.

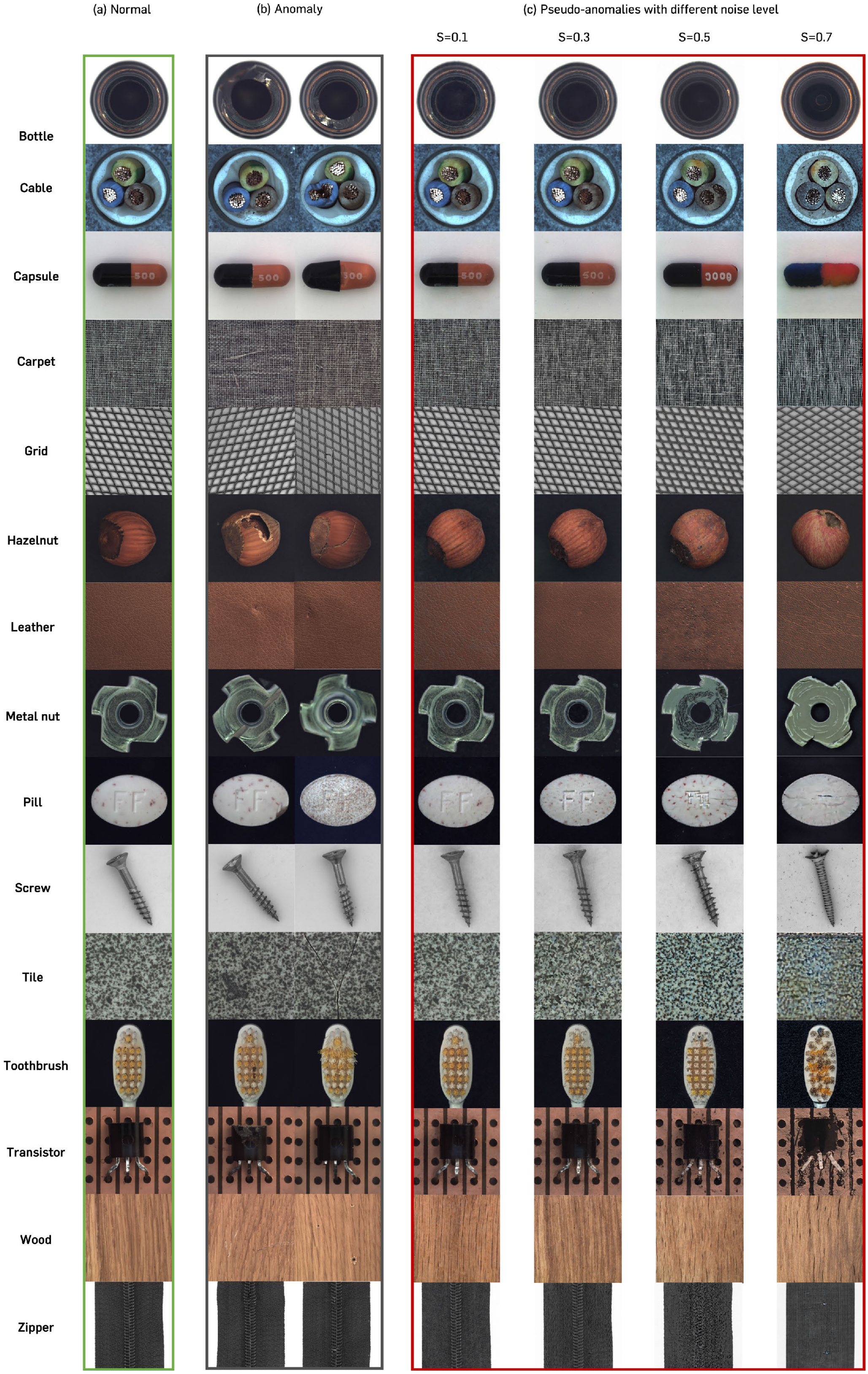

FIGURE 8. Visualization of (a) normal, (b) anomaly and (c) pseudo-anomalies synthesized via pre-trained text-to-image diffusion model with different noise level (S) in MVTec-AD. Pseudo-anomalies are generated with **prompts** incorporating $\mathbf{c}^*_a$, such as "a photo of a $\mathbf{c}^*_a$ with damage".

B. QUALITATIVE RESULTS

In Fig. 5-8, we present additional qualitative results of pseudo-anomalies synthesized using the pre-trained text-to-image diffusion model [34] for both VisA [17] and MVTec-AD [16] datasets. We adjust the level of noise added to the reference image, denoted as S. We explore four different noise level, 0.1, 0.3, 0.5 and 0.7. Fig. 5 and Fig. 7 showcase the pseudo-anomalies conditioned with a simple text prompt, as described in Section IV-B, for the VisA and MVTec-AD datasets, respectively. On the other hand, Fig. 6 and Fig. 8 demonstrate the pseudo-anomalies conditioned with a prompt incorporating $\mathbf{c}_a^*$, as described in Section IV-C, for the VisA and MVTec-AD datasets, respectively. Overall, incorporating $\mathbf{c}_a^*$ in the conditioning prompt generates more fine-grained anomalies compared to the simple text prompt.

REFERENCES

[1] P. Bergmann, K. Batzner, M. Fauser, D. Sattlegger, and C. Steger, "Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization," *Int. J. Comput. Vis.*, vol. 130, no. 4, pp. 947–969, Apr. 2022.

[2] N. Cohen and Y. Hoshen, "Sub-image anomaly detection with deep pyramid correspondences," 2020, *arXiv:2005.02357*.

[3] T. Defard, A. Setkov, A. Loesch, and R. Audigier, "PaDiM: A patch distribution modeling framework for anomaly detection and localization," in *Proc. Int. Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2021, pp. 475–489.

[4] C.-L. Li, K. Sohn, J. Yoon, and T. Pfister, "CutPaste: Self-supervised learning for anomaly detection and localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9659–9669.

[5] N.-C. Ristea, N. Madan, R. T. Ionescu, K. Nasrollahi, F. S. Khan, T. B. Moeslund, and M. Shah, "Self-supervised predictive convolutional attentive block for anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 13566–13576.

[6] K. Roth, L. Pemula, J. Zepeda, B. Schölkopf, T. Brox, and P. Gehler, "Towards total recall in industrial anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14298–14308.

[7] V. Zavrtanik, M. Kristan, and D. Skocaj, "DRÆM—A discriminatively trained reconstruction embedding for surface anomaly detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8330–8339.

[8] S. Sheynin, S. Benaim, and L. Wolf, "A hierarchical transformation-discriminating generative model for few shot anomaly detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8475–8484.

[9] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *Proc. Int. Conf. Mach. Learn.*, vol. 80, 2018, pp. 4393–4402.

[10] J. Yi and S. Yoon, "Patch SVDD: Patch-level SVDD for anomaly detection and segmentation," in *Proc. Asian Conf. Comput. Vis.*, vol. 6, 2020, pp. 375–390.

[11] H. Deng and X. Li, "Anomaly detection via reverse distillation from one-class embedding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9727–9736.

[12] D. Gudovskiy, S. Ishizaka, and K. Kozuka, "CFLOW-AD: Real-time unsupervised anomaly detection with localization via conditional normalizing flows," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 1819–1828.

[13] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "GANomaly: Semi-supervised anomaly detection via adversarial training," in *Proc. 14th Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2018, pp. 622–637.

[14] C. Baur, B. Wiestler, S. Albarqouni, and N. Navab, "Deep autoencoding models for unsupervised anomaly segmentation in brain MR images," in *Proc. Int. MICCAI Brainlesion Workshop*. Cham, Switzerland: Springer, Sep. 2018, pp. 161–169.

[15] D. Gong, L. Liu, V. Le, B. Saha, M. R. Mansour, S. Venkatesh, and A. Van Den Hengel, "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1705–1714.

[16] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, "MVTec AD—A comprehensive real-world dataset for unsupervised anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9584–9592.

[17] Y. Zou, J. Jeong, L. Pemula, D. Zhang, and O. Dabeer, "SPot-the-difference self-supervised pre-training for anomaly detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 392–408.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, and J. Clark, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[19] J. Jeong, Y. Zou, T. Kim, D. Zhang, A. Ravichandran, and O. Dabeer, "WinCLIP: Zero-/few-shot anomaly classification and segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 19606–19616.

[20] Y. Ming, Z. Cai, J. Gu, Y. Sun, W. Li, and Y. Li, "Delving into out-of-distribution detection with vision-language representations," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 35087–35102.

[21] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-Or, "An image is worth one word: Personalizing text-to-image generation using textual inversion," 2022, *arXiv:2208.01618*.

[22] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation," 2022, *arXiv:2208.12242*.

[23] N. Kumari, B. Zhang, R. Zhang, E. Shechtman, and J.-Y. Zhu, "Multi-concept customization of text-to-image diffusion," 2022, *arXiv:2212.04488*.

[24] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 10674–10685.

[25] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. G. Lopes, B. K. Ayan, and T. Salimans, "Photorealistic text-to-image diffusion models with deep language understanding," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 36479–36494.

[26] M. Rudolph, B. Wandt, and B. Rosenhahn, "Same same but DifferNet: Semi-supervised defect detection with normalizing flows," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 1906–1915.

[27] C. Huang, H. Guan, A. Jiang, Y. Zhang, M. Spratlin, and Y. Wang, "Registration based few-shot anomaly detection," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 303–319.

[28] J. Yu, Y. Zheng, X. Wang, W. Li, Y. Wu, R. Zhao, and L. Wu, "FastFlow: Unsupervised anomaly detection and localization via 2D normalizing flows," 2021, *arXiv:2111.07677*.

[29] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 6840–6851.

[30] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," 2020, *arXiv:2010.02502*.

[31] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with CLIP latents," 2022, *arXiv:2204.06125*.

[32] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[33] Y. Zhang, H. Jiang, Y. Miura, C. D. Manning, and C. P. Langlotz, "Contrastive learning of medical visual representations from paired images and text," 2020, *arXiv:2010.00747*.

[34] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, "SDEdit: Guided image synthesis and editing with stochastic differential equations," 2021, *arXiv:2108.01073*.

[35] M. Yang, P. Wu, J. Liu, and H. Feng, "MemSeg: A semi-supervised method for image surface defect detection using differences and commonalities," 2022, *arXiv:2205.00908*.

[36] M. Cherti, R. Beaumont, R. Wightman, M. Wortsman, G. Ilharco, C. Gordon, C. Schuhmann, L. Schmidt, and J. Jitsev, "Reproducible scaling laws for contrastive language-image learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 2818–2829.

[37] G. Xie, J. Wang, J. Liu, F. Zheng, and Y. Jin, "Pushing the limits of fewshot anomaly detection in industry vision: Graphcore," 2023, *arXiv:2301.12082*.

[38] Z. Wang, Y. Zhou, R. Wang, T.-Y. Lin, A. Shah, and S. N. Lim, "Few-shot fast-adaptive anomaly detection," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 4957–4970.

[39] J.-C. Wu, D.-J. Chen, C.-S. Fuh, and T.-L. Liu, "Learning unsupervised metaformer for anomaly detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4349–4358.

SANGKYUNG KWAK (Member, IEEE) received the B.S. degree in computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2023, where he is currently pursuing the M.S. degree with the Kim Jaechul Graduate School, advised by Prof. Jinwoo Shin.

JONGHEON JEONG received the B.S. degree in mathematics and computer science and the Ph.D. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2017 and 2023, respectively. He is currently a Postdoctoral Fellow with KAIST. He was with Amazon Web Services Inc. (AWS AI), Seattle, WA, USA, as an Applied Scientist Intern, from 2021 to 2022. He is interested in making deep learning more reliable against various distribution shifts, such as adversarial examples, corruptions, and novelties. He was a recipient of the 2020 Qualcomm Innovation Fellowship Korea.

HANKOOK LEE received the B.S. degree in mathematical science and the M.S. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2016, 2018, and 2022, respectively. He was with KAIST, as a Postdoctoral Researcher, from 2022 to 2023. Since 2023, he has been a Research Scientist with LG AI Research. He has investigated "How to Learn Deep Neural Networks With Limited Human Prior Knowledge." Specifically, his interests include self-supervised learning, transfer learning, data augmentation, and real-world applications with limited labels.

WOOHYUCK KIM received the B.S. degree in mathematical science and electric engineering (double major) from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2022, where he is currently pursuing the M.S. degree with the Kim Jaechul Graduate School of AI, advised by Prof. Jinwoo Shin.

DONGHO SEO received the B.S. degree in electronics and communication engineering from Hanyang University, Ansan, South Korea, and the Ph.D. degree in electronics and communication engineering from Hanyang University, Seoul, South Korea, in 2021. Since May 2021, he has been with the Electronic Warfare Research and Development Research Center, LIG Nex1, South Korea, where he is currently a Research Engineer. His research interests include machine learning-based radar signal detection, classification in electronic warfare systems, and spectrum sensing in cognitive radio networks.

WOOJIN YUN received the B.S. degree in electronics engineering from Hanyang University, Ansan, South Korea, and the M.S. degree in applied artificial intelligence from Hanyang University, Seoul, South Korea, in 2022. Since July 2022, he has been with the Electronic Warfare Research and Development Research Center, LIG Nex1, South Korea, where he is currently a Research Engineer. His research interests include machine learning-based radar signal detection and classification in electronic warfare systems.

WONJIN LEE received the B.S. degree in control and measurement engineering from Korea University, South Korea, in 1999. In 2002, he joined the Electronic Warfare Research and Development Laboratory, LIG Nex1, South Korea. His research interests include digital receiver, digital signal processing, and time synchronization.

JINWOO SHIN received the B.S. degree in mathematics and computer science from Seoul National University, in 2001, and the Ph.D. degree in mathematics from the Massachusetts Institute of Technology, in 2010. After that, he joined the Korea Advanced Institute of Science and Technology (KAIST), Daejoen, South Korea, in Fall 2013, he started to work on the algorithmic foundations of machine learning. He is currently a Endowed Chair Professor (jointly affiliated) with the Kim Jaechul Graduate School of AI and the School of Electrical Engineering, KAIST. During the Ph.D. study, he received the George M. Sprowls Award (for Best MIT CS Ph.D. Theses).