Memorizing Structure-Texture Correspondence for Image Anomaly Detection

Kang Zhou, Jing Li, Yuting Xiao, Jianlong Yang, Jun Cheng, Wen Liu, Weixin Luo, Jiang Liu, *Senior Member, IEEE*, and Shenghua Gao, *Member, IEEE*

***Abstract*— This work focuses on image anomaly detection by leveraging only normal images in the training phase. Most previous methods tackle anomaly detection by reconstructing the input images with an autoencoder (AE)-based model, and an underlying assumption is that the reconstruction errors for the normal images are small, and those for the abnormal images are large. However, these AE-based methods, sometimes, even reconstruct the anomalies well; consequently, they are less sensitive to anomalies. To conquer this issue, we propose to reconstruct the image by leveraging the structure-texture correspondence. Specifically, we observe that, usually, for normal images, the texture can be inferred from its corresponding structure (e.g., the blood vessels in the fundus image and the structured anatomy in optical coherence tomography image), while it is hard to infer the texture from a destroyed structure for the abnormal images. Therefore, a structure-texture correspondence memory (STCM) module is proposed to reconstruct image texture from its structure, where a memory mechanism is used to characterize the mapping from the normal structure to its corresponding normal texture. As the correspondence between destroyed structure and texture cannot be characterized by the memory, the abnormal images would have a larger reconstruction error, facilitating anomaly detection. In this work, we utilize two kinds of complementary structures (i.e., the semantic structure with human-labeled category information and the low-level structure with abundant details), which are extracted by two structure extractors. The reconstructions from the two kinds of structures are fused together by a learned attention weight to get the final reconstructed image. We further feed the reconstructed image into the two aforementioned structure extractors to extract structures. On the one hand, constraining the consistency between the structures extracted from the original input and that from the reconstructed image would regularize the network training; on the other hand, the error between the structures extracted from the original input and that from the reconstructed image can also be used as a supplement measurement to identify the anomaly. Extensive experiments validate the effectiveness of our method for image anomaly detection on both industrial inspection images and medical images.**

***Index Terms*— Image anomaly detection, industrial inspection image analysis, low-level structure, medical image analysis, semantic structure, structure-texture correspondence memory (STCM).**

Manuscript received December 10, 2020; revised April 9, 2021; accepted July 22, 2021. Date of publication August 13, 2021; date of current version June 2, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0100704, in part by NSFC under Grant 61932020, in part by the Science and Technology Commission of Shanghai Municipality under Grant 20ZR1436000, in part by the Guangdong Provincial Department of Education under Grant 2020ZDZX3043, in part by the Shenzhen Natural Science Fund under Grant JCYJ20200109140820699, in part by the Stable Support Plan Program under Grant 20200925174052004, and in part by the "Shuguang Program" supported by the Shanghai Education Development Foundation and the Shanghai Municipal Education Commission. *(Kang Zhou and Jing Li contributed equally to this work.) (Corresponding author: Shenghua Gao.)*

Kang Zhou and Jing Li are with the School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China, also with the Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: zhoukang@shanghaitech.edu.cn; lijing1@shanghaitech.edu.cn).

Yuting Xiao, Wen Liu, and Weixin Luo are with the School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China.

Jianlong Yang is with the School of Biomedical Engineering, Shanghai Jiao Tong University, Shanghai 200030, China.

Jun Cheng is with the Institute for Infocomm Research, Agency for Science, Technology and Research, Singapore 138632.

Jiang Liu is with the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, Guangdong 518055, China, and also with the Cixi Institute of Biomedical Engineering, Chinese Academy of Sciences, Ningbo 315201, China.

Shenghua Gao is with the School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China, also with the Shanghai Engineering Research Center of Intelligent Vision and Imaging, Shanghai 201210, China, and also with the Shanghai Engineering Research Center of Energy Efficient and Custom AI IC, Shanghai 201210, China (e-mail: gaoshh@shanghaitech.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2021.3101403.

Digital Object Identifier 10.1109/TNNLS.2021.3101403

I. Introduction

IMAGE anomaly detection refers to the identification of abnormality by only leveraging normal images in the training phase [1]–[3]. In real scenarios, abnormal samples (e.g., the uncommon defects in industrial inspection images and the uncommon diseases in medical images) are rare and with various possibilities; thus, it is not easy to collect lots of samples with all possible anomalies. Consequently, traditional image classification methods [4]–[6] cannot be directly applied in these scenarios. In contrast, it is relatively easy to collect normal training samples; therefore, people propose to leverage only normal data to train a model for anomaly detection. Recently, anomaly detection has drawn lots of attention for the industrial image analysis [7] and the medical image analysis [8] because of its potential applications in these domains.

Typical anomaly detection methods [9]–[13] usually follow the reconstruction scheme, and we also focus on this scheme in this article. Previous reconstruction-based methods usually use the autoencoder (AE)- or variational AE (VAE)-based model to reconstruct the input for anomaly detection. In these methods, given an image, an encoder maps the image to a latent feature space, and a decoder reconstructs the image based on the latent feature. Then, the image reconstruction errors for normal images are minimized in the training phase. As it is trained with normal samples only, it is assumed that such a model would have smaller reconstruction errors for normal samples and larger reconstruction errors for abnormal samples.

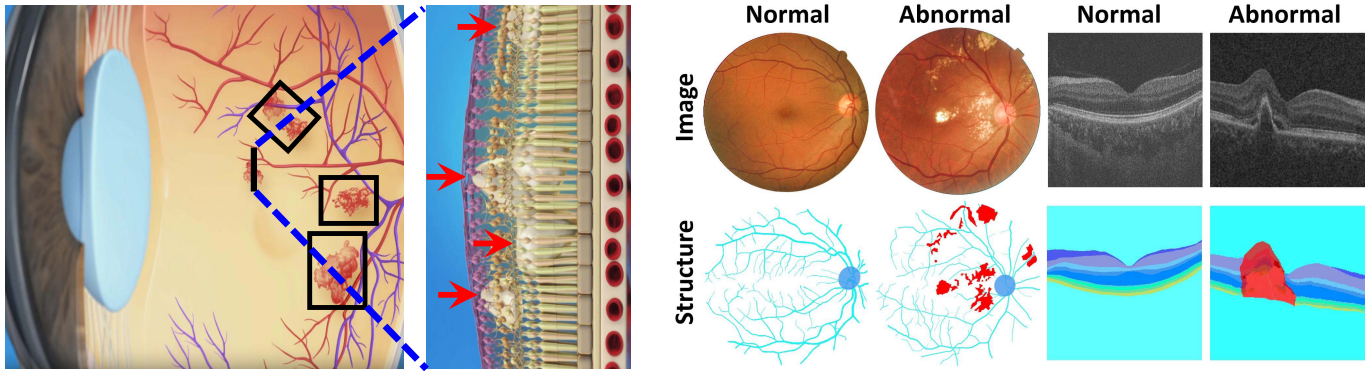

Fig. 1. Motivation of memorizing structure-texture correspondence for image anomaly detection. It is observed that the normal images are highly structured, while the structure is broken by various anomalies in the abnormal images. Therefore, the texture in a normal image can be inferred from the normal structure, while it is hard to infer the abnormal texture from the abnormal structure. For example, the lesions [denoted by the red arrow and black rectangle in (a)] of DR destroy the blood vessel and histology layer in the retina. Thus, in the abnormal retinal fundus image and OCT image, the lesions [denoted by red color in (b) and (c)] broke the structure, leading to the difficulty to infer the abnormal texture from the abnormal structure. We adopt (a) from the website of American Academy of Ophthalmology [16]. (a) Vasculature and histology in retina. (b) Fundus modality. (c) OCT modality.

Therefore, in the test phase, an image can be classified to be normal or abnormal by measuring the image reconstruction error. However, it has been observed that the AE/VAE usually has a small reconstruction error on these abnormal images [14], [15]. Consequently, they are less sensitive to detect abnormal image. It is desired to design a model that is more sensitive to abnormal images.

We observe that the normal image is highly structured, while the regular structure is broken by the disease in abnormal images. Therefore, the normal texture can be inferred from the normal structure, while it is hard to infer the abnormal texture from the abnormal structure, as shown in Fig. 1. Thus, we propose to leverage the structure-texture correspondence for image anomaly detection. In our solution, a structure-texture correspondence memory (STCM) module is proposed, where the mapping from normal structure to its corresponding normal texture is characterized with a memory. As only the correspondence between the normal structure and the normal texture is memorized, a large reconstruction error is reached for the abnormal images, which can be used for better detecting the anomalies.

Concretely, we leverage two types of structures (i.e., the semantic structure and the low-level structure) for reconstruction. We treat the semantically meaningful information (e.g., vessel topological structure in fundus images and the anatomic layer structure in optical coherence tomography (OCT) images) in an image as the semantic structure, but such semantic structure needs expensive human labeling. Usually, such annotations are not provided for anomaly detection data; we propose to use a domain adaptation (DA) approach to leverage the annotation in other annotated datasets to train the semantic structure extraction network. Furthermore, the extracted vessels are usually not complete, and some finer vessels are not included. As a supplement, the low-level structure (e.g., the Canny edge) is abundant of detailed information and does not require additional labeling. As shown in Fig. 2(a), we first extract the two kinds of structures with two structure extractors. The extracted structures are then fed into the STCM module to get two reconstructed images. Then, one image and the other image are fused together to get the final reconstruction image with different weights, which is conditioned on the input image and learned with an attention network, for a better reconstruction for normal images.

In addition, in the scenario of medical image analysis, doctors can make diagnoses with the aid of the structures [17]–[19]. Based on this observation, we fed the reconstructed image into two aforementioned structure extractors again. The structures extracted from the reconstructed image are used from two aspects. In the training phase, the error between the structures extracted from the original input image and that from the reconstructed image is minimized to enforce that the image is well reconstructed, which serves as a regularizer for network training. Meanwhile, a larger error is reached for the abnormal image in the structure space, especially in the semantic structure. Therefore, the semantic structure error can be used as an additional measurement for identifying the anomaly in the test phase.

We term the proposed network as a structure-texture memory network (MemSTC-Net), and the main contributions are summarized as follows.

1) We propose an STCM module by leveraging the correspondence between structure and texture for image reconstruction. Specifically, a memory only stores the mapping from the normal structure to its corresponding normal texture in the STCM module. Therefore, a large reconstruction error is reached for the abnormal image.
2) A complementary pair of structures are leveraged for better reconstruction. To be specific, the semantic structure contains semantically meaningful information, and the low-level structure contains abundant details.
3) Since the semantic structures are not given on most datasets for medical images anomaly detection, we employ a DA method to extract the semantic structure by leveraging other datasets annotated with semantic structure.
4) We encode the reconstructed image back into the structure space, which is used as a regularizer for better normal image reconstruction in the training phase. Meanwhile, besides the error between the reconstructed image and the original input, the error between the semantics structures extracted from them is also utilized to identify the anomaly in the test phase.

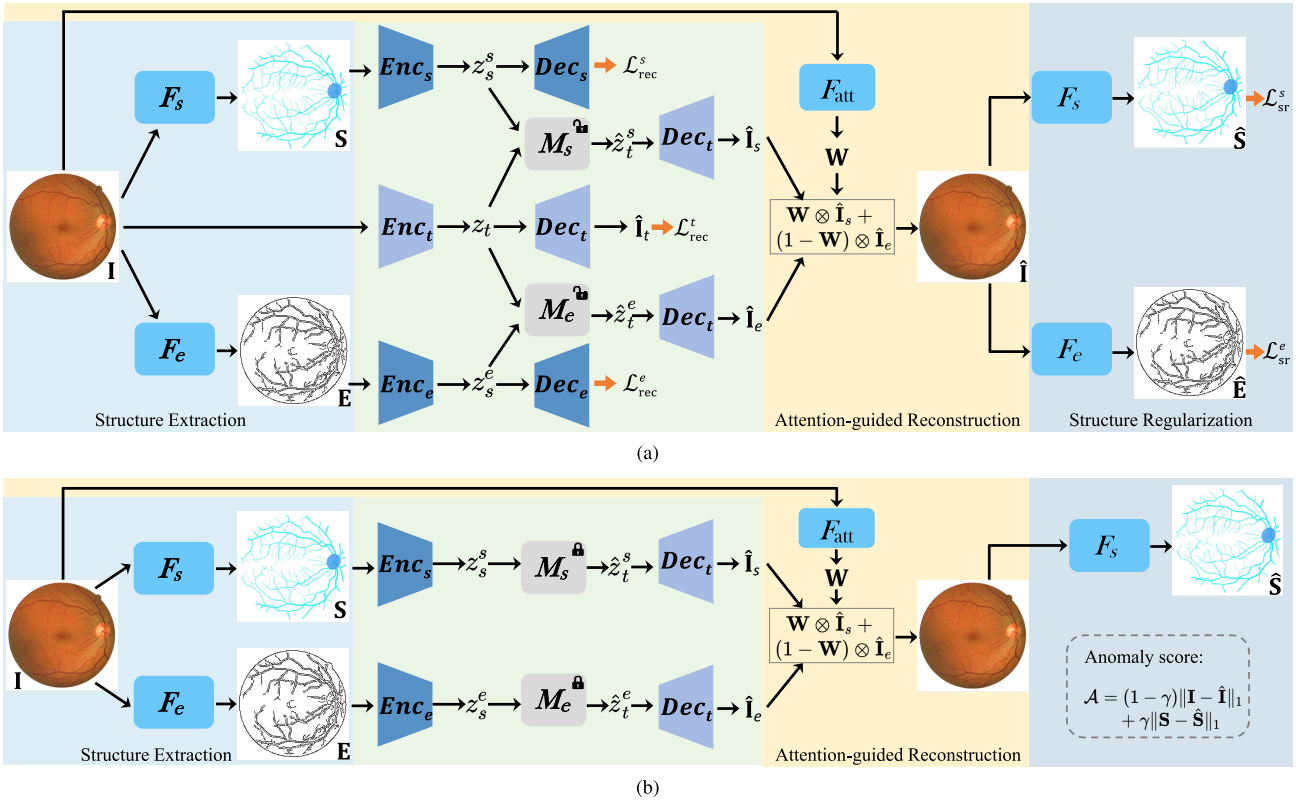

Fig. 2. Overall framework of our MemSTC-Net. We propose to leverage the correspondence between the normal texture and structure to reconstruct the image, where a memory mechanism is used to characterize the mapping from the normal structure feature (z_s^s/z_s^e) to its corresponding normal texture feature ($\hat{z}_t^s/\hat{z}_t^e$). Here, the structure includes semantic structure **S** and low-level structure **E**. (a) Training diagram of our MemSTC-Net. The "open lock" denotes that we update the memory during training. (b) Test diagram of our MemSTC-Net. The "closed lock" denotes that we fix the memory during the test.

This article is an extension of our previous work [20]. We extend the framework in the following aspects.

1) To encode the structure-texture correspondence, in this article, we propose to memorize the correspondence between the structure and its texture. In [20], we encoded the structure-texture relation by fusing the last layer feature in a texture encoder with the structure feature to reconstruct the image. However, since the input of the texture encoder is the original image, the texture encoder probably introduces abnormal information for abnormal image reconstruction in the test phase, which is unfavorable for anomaly detection. Thus, we propose to remove the texture encoder in P-Net [20] and introduces an STCM module to memorize the structure-texture correspondence of the normal images for image reconstruction.
2) We propose to leverage the low-level structure as a complement to the semantic structure, which is the only structure in [20].
3) More experiments are conducted to further validate the effectiveness of our approach.

The rest of this article is organized as follows. In Section II, we introduce the work related to the proposed method. In Section III, we detail our proposed MemSTC-Net for image anomaly detection. In Section IV, extensive experiments are conducted to validate the effectiveness of our method. We conclude our work in Section V.

II. Related Work

Anomaly detection usually refers to one-class learning, which is essentially a semisupervised case that is trained only with normal data in an inductive learning manner. In such a case, the training data are not polluted by anomalies, and the goal is to detect whether a test sample is abnormal or not [21], [22]. Differently, outlier detection refers to training a model from the unlabeled polluted data, which contains both normal data and abnormal data [21]–[23]. As the outliers make it harder to fit the model, the goal of outlier detection is to remove the outliers during the training in a transductive learning manner [23], [24]. In some literature works, these two terms are used interchangeably [1]–[3], [25]. In this article, we focus on anomaly detection.

A. Anomaly Detection

Anomaly detection is a valuable field in the machine learning community [1], [3]. In the anomaly detection problem, the anomalies are defined as the samples out of the distribution of the normal ones. It is natural to learn a discriminative hyperplane to separate the abnormal samples from the normal samples. The one-class support vector machine (OCSVM) [26] and the kernel density estimation [27] are two delegates of the classical anomaly detection methods. However, these methods often fail in the scenarios where the data are high-dimensional and large scale due to the expensive computational cost [28]. Therefore, Ruff *et al.* [28] proposed a deep one-class SVDD that trains a neural network while minimizing the volume of a hypersphere that encloses the network representations of the data. In this way, the potential anomalies are far away from the hypersphere center. Besides, Li *et al.* [29] proposed to use the Gaussian mixture models (GMMs) to model the distribution of normal samples, and the samples out of the

mixed Gaussian distribution are probably abnormal. Based on GMM, Zong *et al.* [15] proposed to utilize a deep AE to generate a low-dimensional feature and reconstruction error for each input sample, which are further fed into a GMM. Since the GMM has many parameters, McNicholas and Murphy [30] proposed to reduce the parameters by a latent Gaussian model, which is closely related to the factor analysis model (a data reduction technique) and yielded parsimonious mixture models (PMMs). When the feature space is large, clusters may manifest anomalies on the very small feature subsets, which can be well-captured by the PMM. In this way, Miller *et al.* [31] proposed a method with PMM for anomaly detection, which is used for both the null and the alternative hypothesis and with the Bayesian information criterion adjudicating between these hypotheses. Ionescu *et al.* [32] and Yu *et al.* [33] use a deep AE to learn the low-dimensional features of input data, and the clustering-based anomaly measures are used in the latent representation space. Pang *et al.* [34] proposed a ranking model-based framework, which optimizes the representations so that the nearest neighbor distance of pseudolabeled anomalies is larger than that of pseudolabeled normal instances.

In the image anomaly detection areas, Carrera *et al.* [35] proposed to use convolutional sparse models to learn a dictionary of filters to detect abnormal regions. AnoGAN is proposed by Schlegl *et al.* [8], which introduced generative adversarial network (GAN) [36] to generate normal images from a latent space with Gaussian distribution in the training phase, and test samples are recognized as anomalies when the corresponding latent code is out of the distribution. In [8], the residual loss is introduced to map the image to the latent space, but this process is slow. To address this issue, Schlegl *et al.* [37] proposed to use an encoder to learn the mapping from the image to the latent space. Similar to AnoGAN [8], GANomaly proposed by Akcay *et al.* [38] also involved representation learning in a latent space, which trains an encoder–decoder–encoder network with the adversarial learning scheme to capture the normal distribution from both images and latent space. Zimmerer *et al.* [12] proposed the context-encoding VAE for anomaly detection on brain MRI, while Chen and Konukoglu [10] initially proposed to use adversarial AE for anomaly detection on brain MRI. Almost at the same time, Baur *et al.* [9] proposed to use a deep AE that combines spatial AEs and GANs for anomaly detection on brain MRI. As discussed before, the structure and texture in normal images are closely related. However, these existing methods do not exploit encoding the structure-texture correspondence for image anomaly detection. More works related to anomaly detection can be found in the recent comprehensive survey paper [2], [3].

B. Outlier Detection

Outlier detection is usually used for data cleaning: removing the outliers from the training set such that the desired parametric statistical model can fit the data more smoothly [23]. Before the deep learning era, the statistical model, the neighbor-based method, and the method based on the principal component analysis (PCA) are applied to outlier detection [39]–[41]. Specifically, the statistical method fits the distributions on data [39], and the outliers have a lower probability than the inliers under the learned distributions. The neighbor-based method assumes that the inliers have dense areas, while the outliers are far from these areas [40]. Xu *et al.* [41] learn the PCA projections from the data, and the samples with the largest variances are identified as outliers. In the deep learning era, Xia *et al.* [24] addressed outlier detection by leveraging the reconstruction error of an AE and its variant. They showed that an AE is simple but effective for outlier detection. Furthermore, they gradually inject discriminative information in the learning process to make the inliers and outliers more separable. Instead of the commonly used AE in previous methods, Wang *et al.* [42] proposed an E^3Outlier framework in an effective and end-to-end manner. Specifically, E^3Outlier leverages a discriminative network with the self-supervised learning for better feature representation. Wang *et al.* [42] also exploit a novel inlier priority to enable end-to-end framework by the discriminative network. Zong *et al.* [15] proposed a deep autoencoding GMM, and they conducted experiments under both anomaly detection and outlier detection problem.

C. Structure-Texture Relation Encoding Networks

Image structure has been successful used for semantic image synthesis [43], image inpainting [44], [45], and video inpainting [46]. Tang *et al.* [43] proposed to use the edge as an intermediate representation since the edge introduces detailed structure information, which is further adopted to guide the texture generation. The edge information is also used in the image inpainting task. Nazeri *et al.* [45] proposed a model, in which the full edge map of an incomplete image is predicted by an edge generator. Then, the predicted edge map and the incomplete image are concatenated and fed to an image completion network to inpaint the full image with generated texture. However, the intensity distribution of the edge map is notably different from that of the image. To address this issue, Ren *et al.* [44] proposed to leverage an edge-preserved smooth image to represent the structure in an image and proposed a two-stage model that splits the inpainting task into two parts: structure reconstruction and texture generation. In video inpainting, Wang *et al.* [46] proposed to first complete edges in the missing regions via an edge inpainting network with 3-D convolutions networks. Then, the proposed method reconstructs the textures using a coarse-to-fine synthesis network under the guidance of the predicted edges. Among these methods, the relation of structure texture is implicitly learned by the neural network in the "original image-to-predicted structure-to-reconstructed image" pipeline. Motivated by these methods, we propose to model the normal structure-texture relation for image anomaly detection. Besides, the structure-texture correspondence is modeled explicitly in the proposed STCM.

D. Memory-Augmented Networks for Anomaly Detection

Memory-augmented networks are neural networks that have external memory where the information can be saved and loaded. Memory-augmented networks have attracted interest to solve anomaly detection problems [14], [47], [48]. Gong *et al.* [14] introduced a memory-augmented AE (MemAE) for anomaly detection in natural images and surveillance videos. Specifically, the memory in [14] is designed by recording the prototypical patterns in normal images or videos. Since the patterns in natural images and videos are diverse, Gong *et al.* [14] propose to use a soft addressing vector for accessing the memory and apply a hard shrinkage operation to promote the sparsity of the addressing vector. Based on the MemAE [14], Zhang *et al.* [47] proposed a memory-augmented anomaly GAN (MA-GAN) that improves the MemAE with a discriminator [36], for retinal OCT screening. For anomaly detection in diverse images and videos,

the main drawback of MemAE [14] is that it does not consider the diversity of normal patterns explicitly. To address this problem, Park *et al.* [48] proposed to use a memory module with a new update scheme where items in the memory record prototypical patterns of normal data. As a summary, previous methods adopted a memory to memorize the latent feature of normal images for image anomaly detection. However, this strategy is unsuitable to explicitly encode the structure-texture correspondence. To memorize the structure-texture correspondence, we create a memory to save the paired information of structure and its texture, and we use the structure feature as a query to retrieve the corresponding texture feature.

E. Self-Supervision for Anomaly/Outlier Detection

Recently, many self-supervised learning methods are proposed to learn the general image features from unlabeled data without any human annotations, which can be used for downstream tasks, such as image classification and segmentation [49]. The self-supervision signal includes image rotation [50], image jigsaw puzzle [51], image patch permutation [52], gray image colorizing [53], image inpainting [54], and so on. Recently, some literature [12], [42], [55], [56] explores self-supervision for anomaly/outlier detection. Specifically, Golan and El-Yaniv [55] proposed to train a multiclass model to discriminate the types of geometric transformation applied to the normal images. At test time, the distribution of softmax response values of training images is used to detect anomalies. Inspired by the success of inpainting for feature learning [54], Zimmerer *et al.* [12] proposed a novel image reconstruction network, which takes the image with a missing region as input and takes the original image as ground truth for reconstruction. Similarly, the method proposed (termed SMAI) in [56] also aims to learn features with inpainting for anomaly detection. Differently, SMAI [56] first performs superpixel segmentation on the input images, and then, SMAI trains an inpainting module on the normal samples through random superpixel masking and restoration. Thus, the model can fill the superpixel mask with normal content in reconstruction. During the test phase, SMAI masks the image using superpixels and restores them one by one. In this way, SAMI can identify the abnormal regions by comparing the mask areas of the original images and their reconstruction. For outlier detection, Wang *et al.* [42] proposed to create multiple pseudoclasses by various simple operations on the original unlabeled data. These operations include regular affine transformation, irregular affine transformation, and patch rearranging. Differently, in our method, we leverage the cross-modality for reconstruction, which contains the mapping between the image and its low-level structure. That is to say, our method can be regarded as a type of self-supervision, where the low-level structure is leveraged as an intermediate self-supervised representation for image reconstruction.

III. METHOD

In this article, we propose an STCM network (MemSTC-Net) for image anomaly detection, which leverages the correspondence between the image texture and structure to reconstruct the image. Specifically, as shown in Fig. 2, the proposed network consists of five modules.

1) Structure extraction module, where two networks extract semantic structure $\mathbf{S}$ and low-level structure $\mathbf{E}$ from the input image $\mathbf{I}$, respectively.
2) Structure and texture feature encoding module, which encodes the feature of the texture and two kinds of the structure by $\boldsymbol{Enc}_s$, $\boldsymbol{Enc}_e$, and $\boldsymbol{Enc}_t$, respectively. This module also contains the decoders $\boldsymbol{Dec}_s$ and $\boldsymbol{Dec}_e$ to map the structure feature (z_s^s and z_s^e) to the reconstructed structure.
3) STCM module, in which the structure-texture correspondence is memorized, and this module produces the image $\hat{\mathbf{I}}_s$ reconstructed from the semantic structure and the image $\hat{\mathbf{I}}_e$ reconstructed from the low-level structure by the texture decoder $\boldsymbol{Dec}_t$. To memorize the structure-texture correspondence for semantic structure and low-level structure, two memory blocks ($\boldsymbol{M}_s$ and $\boldsymbol{M}_e$) are introduced.
4) Attention-guided fusion module, which fuses $\hat{\mathbf{I}}_s$ and $\hat{\mathbf{I}}_e$ with a learned attention weight automatically.
5) Structure regularization (SR) module, which further extracts semantic structure $\hat{\mathbf{S}}$ and low-level structure $\hat{\mathbf{E}}$ from the reconstructed image $\hat{\mathbf{I}}$. By minimizing the difference between $\mathbf{S}$ and $\hat{\mathbf{S}}$ and the difference between $\mathbf{E}$ and $\hat{\mathbf{E}}$, this module enforces the original image to be correctly reconstructed.

A. Structure Extraction Module

In this article, we define two types of structure: the semantic structure, e.g., the vessel in retinal fundus image and the anatomical layer in retinal OCT, and the low-level structure, e.g., the edge, and leverage them together for a better reconstruction for normal images. We pretrain two convolution neural networks as the structure extractors to extract the two kinds of structure. Once the structure extractors are well trained, we fix the module to simplify the optimization of the other modules in our MemSTC-Net. Since there is no semantic annotation in industrial inspection images [7], we only use the low-level structure for industrial inspection images.

1) Semantic Structure: The semantic structure is with semantically meaningful category information but needs expensive human labeling, loses detailed information, and, sometimes, is inaccurate and incomplete. We detect the anomaly in two medical image datasets, i.e., a retinal fundus dataset [57] and a retinal OCT image dataset [58]. However, the semantic structures in both datasets are not provided. Fortunately, there are several publicly available datasets for vessel segmentation in retinal fundus images and layer segmentation in retinal OCT images [59]–[61]. To leverage the existing annotations in the publicly available datasets and overcome the domain shift issue, e.g., different datasets that have different noises and data distribution caused by various devices, we propose to use AdaptSegNet [62], a DA-based semantic segmentation method, to learn the semantic structure extractor.

Specifically, as illustrated in Fig. 3, we map the image in different datasets but with the same modality to their corresponding semantic structure with a U-Net [63] and add a discriminator to make the segmentation results from source and target datasets indistinguishable. For the fundus image, we use the DRIVE dataset [60] as the source, while, for the OCT image, we use the Topcon dataset [61] as the source. The training loss to train the semantic structure extractor $\boldsymbol{F}_s$ is given as follows:

$$\mathcal{L}_{\text{src}} = -\sum \mathbf{S}_{\text{src}} \log(\boldsymbol{F}_s(\mathbf{I}_{\text{src}})) \tag{1}$$

$$\mathcal{L}_{\text{tar}} = \mathbb{E}[\log(1 - D_{\text{da}}(\boldsymbol{F}_s(\mathbf{I}_{\text{tar}}))] + \mathbb{E}[\log D_{\text{da}}(\boldsymbol{F}_s(\mathbf{I}_{\text{src}}))] \tag{2}$$

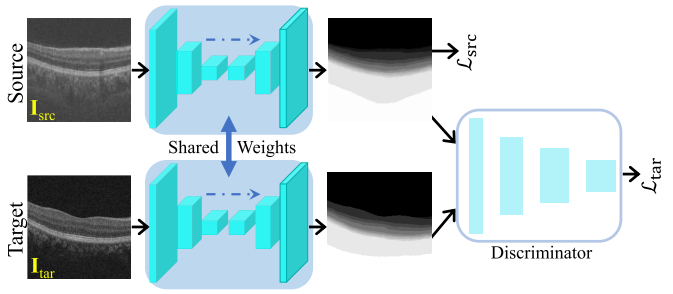

Fig. 3. Training diagram of the semantic structure extraction network $\boldsymbol{F}_s$ with DA.

where $\mathbf{S}_{\text{src}}$ and $\mathbf{I}_{\text{src}}$ denote the ground truth and the source image, respectively. $\mathbf{I}_{\text{tar}}$ denotes the target image, and D_{da} denotes the discriminator in AdaptSegNet [62].

2) Low-Level Structure: To be a complement of the semantic structure, the low-level structure does not require additional human labeling and has abundant details but, sometimes, is redundant for reconstruction. To be specific, we take the Canny edge as the low-level structure. However, the Canny edge detector [64] is not differentiable; thus, we train a network $\boldsymbol{F}_e$ to extract the edge map from the input image with the supervision of the Canny edge detector. The training loss to train the low-level structure extractor $\boldsymbol{F}_e$ is given as follows:

$$\mathcal{L}_{\text{edge}} = -\sum \mathbf{E} \log(\boldsymbol{F}_e(\mathbf{I})) \tag{3}$$

where $\mathbf{I}$ and $\mathbf{E}$ denote the input image and its ground truth, respectively.

B. Structure and Texture Feature Encoding Module

To encode the feature of the texture and two kinds of structure for subsequent memorization and image reconstruction, we use the encoder–decoder to learn the feature in an unsupervised manner. As shown in Fig. 2(a), to learn a representative texture feature z_t with an unsupervised manner, the encoded texture feature z_t from the texture encoder $\boldsymbol{Enc}_t$ is fed to the texture decoder $\boldsymbol{Dec}_t$ to get the reconstructed image $\hat{\mathbf{I}}_t$. Then, we minimize the $L1$-norm of the image reconstruction error as $\mathcal{L}_{\text{rec}}^t = \|\mathbf{I} - \hat{\mathbf{I}}_t\|_1$. Similarly, the encoded structure feature z_s^s/z_s^e from the structure encoder $\boldsymbol{Enc}_s/\boldsymbol{Enc}_e$ is inputted to the structure decoder $\boldsymbol{Dec}_s/\boldsymbol{Dec}_e$ to reconstruct the corresponding structure, respectively. Then, we also minimize $L1$-norm of the structure reconstruction error to learn the structure feature z_s^s and z_s^e with an unsupervised manner. To be specific, the structure reconstruction losses $\mathcal{L}_{\text{rec}}^s$ and $\mathcal{L}_{\text{rec}}^e$ are for the semantic structure and the low-level structure, respectively. Therefore, we get the overall loss function of this module

$$\mathcal{L}_{\text{enc}} = \lambda_1 \mathcal{L}_{\text{rec}}^t + \lambda_2 \mathcal{L}_{\text{rec}}^s + \lambda_3 \mathcal{L}_{\text{rec}}^e \tag{4}$$

where $\lambda_1 = 0.5$, $\lambda_2 = 0.5$, and $\lambda_3 = 0.5$ are the hyperparameters.

C. Structure-Texture Correspondence Memory Module

It is natural for humans to infer the texture of a normal image from the corresponding normal structure. Motivated by this, we propose to memorize the correspondence between the normal structure and its texture to reconstruct the image. Since the memory only stores the normal structure-texture correspondence, and the memory is only updated in the training phase, it is expected that the stored correspondence does not work for the abnormal images during the test, leading to a large reconstruction error, which favors the anomaly detection. Specifically, the memory stores the correspondence as the key–value pairs, where the structure feature is the key, while the texture feature is the value. Both the structure feature $\tilde{z}_s$ and the texture feature $\tilde{z}_t$ are from the same pixel. The key–value pairs stored in the memory can be described as

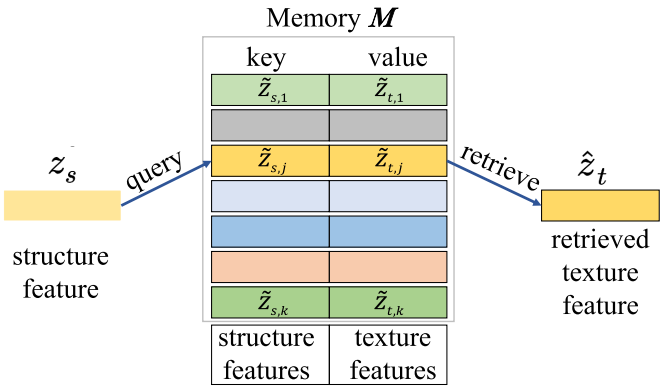

Fig. 4. Illustration of the memory $\boldsymbol{M}$. Different colors denote different key–value pairs.

$$\boldsymbol{M} = \{(\tilde{z}_{s,1}, \tilde{z}_{t,1}), \ldots, (\tilde{z}_{s,j}, \tilde{z}_{t,j}), \ldots, (\tilde{z}_{s,k}, \tilde{z}_{t,k})\} \tag{5}$$

where k is the memory size. As illustrated in Fig. 4, for a query structure feature z_s extracted from the previous structure and texture feature encoding module, the texture feature $\hat{z}_t$ is retrieved for subsequent image reconstruction according to the nearest key structure feature $\hat{z}_s$ in the Euclidean distance as

$$\hat{z}_t = \tilde{z}_{t,\text{J}}, \text{ where } \text{J} = \arg\min_j \|z_s - \tilde{z}_{s,j}\|_2, \quad j \in \{1, \ldots, k\}. \tag{6}$$

To handle the semantic structure feature z_s^s and the low-level structure feature z_s^e from the structure and texture feature encoding module independently, two memories $\boldsymbol{M}_s$ and $\boldsymbol{M}_e$ are instantiated. $\boldsymbol{M}_s$ memorizes the correspondence between the semantic structure feature and the texture feature z_t, while the correspondence between the low-level structure feature and the texture feature z_t is memorized in $\boldsymbol{M}_e$.

Memory Update: To update the memory, we adopt a simple but effective strategy, i.e., the first-in–first-out (FIFO) algorithm [65]. In the FIFO algorithm, the first item that arrives at the memory is the first one to be updated.

Specifically, in the minibatch training, let n denote the batch size, and $n \ll k$. We denote $(z_{s,i}, z_{t,i})$ as the ith structure feature and texture feature in a batch, where $i \in \{1, 2, \ldots, n\}$. The memory block is updated as follows:

$$(\tilde{z}_{s,j}, \tilde{z}_{t,j}) \leftarrow \begin{cases} (\tilde{z}_{s,j-n}, \tilde{z}_{t,j-n}), & j > n \\ (z_{s,j}, z_{t,j}), & j \le n \end{cases} \tag{7}$$

where "$\leftarrow$" denotes the update operation.

D. Attention-Guided Fusion Module

After the retrieval of the two texture features from the memory, we feed them to the same texture decoder in Section III-B to get two reconstructed images, i.e., $\hat{\mathbf{I}}_s$ from the semantic structure and $\hat{\mathbf{I}}_e$ from the low-level structure. To get the final

reconstructed image $\hat{\mathbf{I}}$, a simple solution is to average $\hat{\mathbf{I}}_s$ and $\hat{\mathbf{I}}_e$ as $\hat{\mathbf{I}} = (\hat{\mathbf{I}}_s + \hat{\mathbf{I}}_e)/2$. However, the effect of semantic structure and low-level structure for the final reconstructed image may be distinctive among different input images. Therefore, we propose to learn an attention weight $\mathbf{W}$ by the attention network $\boldsymbol{F}_{\text{att}}$, which is conditioned on the input image $\mathbf{I}$. Then, we use the learned attention weight to fuse $\hat{\mathbf{I}}_s$ and $\hat{\mathbf{I}}_e$ to get the final reconstructed image. The generation of the attention weight $\mathbf{W}$ can be described as

$$\mathbf{W} = \sigma[\boldsymbol{F}_{\text{att}}(\mathbf{I})] \tag{8}$$

where σ is a `sigmoid` function to normalize the attention weight to [0, 1]. After obtaining the learned attention weight, the final reconstructed image $\hat{\mathbf{I}}$ is obtained with the attention weight as

$$\hat{\mathbf{I}} = \mathbf{W} \otimes \hat{\mathbf{I}}_s + (1 - \mathbf{W}) \otimes \hat{\mathbf{I}}_e \tag{9}$$

where $\otimes$ denotes the elementwise multiplication. To compute the reconstruction loss, following [38] and [66], we use $\mathcal{L}_1$ norm to measure the difference between the reconstructed image and the original image as follows:

$$\mathcal{L}_{\text{rec}} = \|\mathbf{I} - \hat{\mathbf{I}}\|_1. \tag{10}$$

To further improve the quality of the reconstructed image, we introduce a discriminator D from GAN [36], [66] to penalize the reconstruction error for the reconstructed image $\hat{\mathbf{I}}$. The adversarial loss $\mathcal{L}_{\text{adv}}$ for training the reconstruction network is

$$\mathcal{L}_{\text{adv}} = \mathbb{E}[\log(1 - D(\hat{\mathbf{I}}))] + \mathbb{E}[\log D(\mathbf{I})]. \tag{11}$$

By minimizing $\mathcal{L}_{\text{adv}}$, the reconstruction network can be trained. To update discriminator, we follow [36] and [66] and maximize $\mathcal{L}_{\text{adv}}$.

E. Structure Regularization Module

We further append the structure extractors $\boldsymbol{F}_s$ and $\boldsymbol{F}_e$ from Section III-A on the reconstructed image as the structure regularizers. The structure regularizers are introduced to enforce the structure extracted from the original image and that from the reconstructed image to be the same; in this way, the original image can be better reconstructed. The loss functions in this module are defined as follows:

$$\begin{aligned} \mathcal{L}^s_{\text{sr}} &= \|\mathbf{S} - \hat{\mathbf{S}}\|_1 \\ \mathcal{L}^e_{\text{sr}} &= \|\mathbf{E} - \hat{\mathbf{E}}\|_1. \end{aligned} \tag{12}$$

The error between the semantic structures from the original image and the reconstructed image is also used for normality measurement in Section III-G.

F. Training and Objective Function

As aforementioned in Section III-A, we first train the semantic structure extractor $\boldsymbol{Enc}_s$ and the low-level structure extractor $\boldsymbol{Enc}_e$, respectively. As shown in Fig. 2(a), we compute $\mathcal{L}_{\text{enc}} = \lambda_1\mathcal{L}^t_{\text{rec}} + \lambda_2\mathcal{L}^s_{\text{rec}} + \lambda_3\mathcal{L}^e_{\text{rec}}$ to learn the feature of structure and texture to update the memory. Thus, we arrive at the objective function of our method to train the structure and texture feature encoding module and the attention-guided fusion module

$$\mathcal{L} = \mathcal{L}_{\text{enc}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{adv}}\mathcal{L}_{\text{adv}} + \underbrace{\lambda^s_{\text{sr}}\mathcal{L}^s_{\text{sr}} + \lambda^e_{\text{sr}}\mathcal{L}^e_{\text{sr}}}_{\text{structure regularization}} \tag{13}$$

where λ_{rec}, λ_{adv}, λ^s_{sr}, and λ^e_{sr} are the hyperparameters. We analyze the effect of these hyperparameters in Section IV-C5 (hyperparameters analysis).

G. Anomaly Detection on Test Data Considering Structure

Previous methods detect the anomaly by the image error [9]–[12] as

$$\mathcal{A}_{\text{img}} = \|\mathbf{I} - \hat{\mathbf{I}}\|_1. \tag{14}$$

A higher image error $\mathcal{A}_{\text{img}}$ for the test image indicates that it is more likely to be abnormal. However, we argue that the structure (especially the semantic structure) is beneficial for detecting the anomaly rather than solely relying on the image, as illustrated in Fig. 1. We define the semantic structure error as

$$\mathcal{A}_{\text{struct}} = \|\mathbf{S} - \hat{\mathbf{S}}\|_1. \tag{15}$$

As illustrated in Fig. 2(b), we consider the image error $\mathcal{A}_{\text{img}}$ and the semantic structure error $\mathcal{A}_{\text{struct}}$ jointly for anomaly detection as

$$\begin{aligned} \mathcal{A} &= (1 - \gamma)\mathcal{A}_{\text{img}} + \gamma\,\mathcal{A}_{\text{struct}} \\ &= (1 - \gamma)\|\mathbf{I} - \hat{\mathbf{I}}\|_1 + \gamma\,\|\mathbf{S} - \hat{\mathbf{S}}\|_1 \end{aligned} \tag{16}$$

where γ is a hyperparameter used to balance the image error and the semantic structure error for anomaly measurement. The reason for not considering the low-level structure error is that the low-level structure is noisy in the background regions, which may be unfavorable for anomaly detection.

IV. EXPERIMENTS

A. Experimental Setup

In this section, we introduce the network architectures, training details, and evaluation metrics.

1) Network Architectures: Let C_k denote a Convolution-BatchNorm-ReLU layer with k filters. All convolutional operations are 3×3 spatial filters applied with stride 1, which will produce a feature map that has the same spatial size as the input feature map. The scale factor of both the max-pooling layer (denoted as Mp) in the encoder and the upsampling layer (denoted as Up) in the decoder is 2.

The network architectures of $\boldsymbol{Enc}_s$, $\boldsymbol{Enc}_e$, and $\boldsymbol{Enc}_t$ are the same. Similarly, the network architectures of $\boldsymbol{Dec}_s$, $\boldsymbol{Dec}_e$, and $\boldsymbol{Dec}_t$ are the same. All of the encoders contain four max-pooling layers, while all of the decoders contain four upsampling layers. Specifically, the encoder architecture is $C_{64} - C_{64} - \text{Mp} - C_{128} - C_{128} - \text{Mp} - C_{256} - C_{256} - \text{Mp} - C_{512} - C_{512} - \text{Mp} - C_{512} - C_{512}$, and the decoder architecture is $\text{Up} - C_{512} - C_{512} - \text{Up} - C_{256} - C_{256} - \text{Up} - C_{128} - C_{128} - \text{Up} - C_{64} - C_{64} - C_m$, where m denotes the channel number of original image. The network architecture of $\boldsymbol{F}_{\text{att}}$ is $C_{64} - C_{64} - \text{Mp} - C_{128} - C_{128} - \text{Up} - C_{64} - C_1$. The network architectures of $\boldsymbol{F}_s$ and $\boldsymbol{F}_e$ are the U-Net [63], and the implementation details of $\boldsymbol{F}_s$ and $\boldsymbol{F}_e$ follow [67].

2) Training Details: To train the proposed network, the batch size is 8, and the input image size is 224×224. The optimizers for the generator and the discriminator are both Adam [68]; we set the learning rate to 0.001, $\beta_1 = 0.1$, and $\beta_2 = 0.9$. In the memory, we set the memory size $k = 2048$. We train our model for 800 epochs. We implement our method with the PyTorch [69] on the NVIDIA TITAN V GPU.

TABLE I

FIVE CATEGORIES AT THE TOP OF THE TABLE ARE TEXTURES (CARPET, GRID, LEATHER, TILE, AND WOOD) IMAGE, AND THE OTHER TEN CATEGORIES AT THE BOTTOM OF THE TABLE ARE OBJECTS (BOTTLE, CABLE, CAPSULE, HAZELNUT, AND SO ON) IMAGE. FOR EACH CATEGORY, THE TOP ROW IS THE ANOMALY REGION **OVERLAP**, WHICH IS THE SAME AS THE EVALUATION METRIC IN [7] AND THE BOTTOM ROW IS **AUC**

Categories	AE (SSIM)	AE (L2)	Ano GAN	CFD	Deep SVDD	Cycle GAN	VAE-GAN	GAN omaly	TI	P-Net	Our Method
Carpet	**0.69**	0.38	0.34	0.20	-	0.04	0.01	0.23	0.29	0.14	0.28
	0.87	0.59	0.54	0.72	0.54	0.46	0.35	0.55	**0.88**	0.57	0.61
Grid	**0.88**	0.83	0.04	0.02	-	0.36	0.04	0.41	0.01	0.59	0.65
	0.94	0.90	0.58	0.59	0.59	0.86	0.76	0.80	0.72	0.98	**0.99**
Leather	0.71	0.67	0.34	0.74	-	0.09	0.12	0.31	**0.98**	0.52	0.63
	0.78	0.75	0.64	0.87	0.73	0.65	0.64	0.77	**0.97**	0.89	0.87
Tile	0.04	0.23	0.08	0.14	-	0.14	0.09	0.19	0.11	0.23	**0.27**
	0.59	0.51	0.50	0.93	0.81	0.64	0.70	0.69	0.41	0.97	**0.98**
Wood	0.36	0.29	0.14	**0.47**	-	0.19	0.11	0.32	0.51	0.37	0.44
	0.73	0.73	0.62	0.91	0.87	0.95	0.77	0.91	0.78	**0.98**	**0.98**
Bottle	0.15	0.22	0.05	0.07	-	0.09	0.11	0.13	-	0.43	**0.45**
	0.93	0.86	0.86	0.78	0.86	0.76	0.73	0.82	-	**0.99**	0.97
Cable	0.01	0.05	0.01	0.13	-	0.02	0.05	0.14	-	0.16	**0.25**
	0.82	**0.86**	0.78	0.79	0.71	0.61	0.60	0.83	-	0.70	0.81
Capsule	0.09	0.11	0.04	0.00	-	0.04	0.19	0.51	-	**0.64**	0.58
	0.94	0.88	0.84	0.84	0.69	0.61	0.59	0.72	-	0.84	0.87
Hazelnut	0.00	0.41	0.02	0.00	-	0.33	0.34	0.37	-	0.66	**0.73**
	0.97	0.95	0.87	0.72	0.71	0.87	0.75	0.86	-	0.97	**0.98**
Metal Nut	0.01	**0.26**	0.00	0.13	-	0.04	0.01	0.18	-	0.24	0.22
	0.89	0.86	0.76	0.82	0.75	0.43	0.46	0.69	-	0.79	0.82
Pill	0.07	0.25	0.17	0.00	-	0.29	0.01	0.17	-	**0.58**	0.54
	0.91	0.85	0.87	0.68	0.77	0.80	0.62	0.76	-	**0.91**	0.87
Screw	0.03	0.34	0.01	0.00	-	0.17	0.02	0.24	-	0.32	**0.41**
	0.96	0.96	0.80	0.87	0.64	0.95	0.97	0.72	-	**1.00**	0.99
Toothbrush	0.08	0.51	0.07	0.00	-	0.13	0.10	0.48	-	**0.63**	0.59
	0.92	0.93	0.90	0.77	0.70	0.70	0.67	0.82	-	0.99	**1.00**
Transistor	0.01	0.22	0.08	0.03	-	0.20	0.05	0.15	-	0.24	**0.27**
	0.90	0.86	0.80	0.66	0.65	0.72	0.78	0.79	-	0.82	0.89
Zipper	0.10	0.13	0.01	0.00	-	0.05	0.04	0.21	-	0.34	**0.38**
	0.88	0.77	0.78	0.76	0.74	0.63	0.60	0.84	-	0.90	**0.93**
Mean	0.22	0.33	0.09	0.13	-	0.15	0.09	0.27	-	0.41	**0.45**
	0.87	0.82	0.74	0.78	0.72	0.71	0.66	0.77	-	0.89	**0.90**

3) Evaluation Metric: Following previous work [8], [70], we calculate the area under receiver operation characteristic (AUC), for normal/abnormal classification by gradually changing the threshold of $\mathcal{A}$, as introduced in Section III-G. A higher AUC indicates that the performance of the method is better.

B. Anomaly Detection on the Industrial Inspection Images

1) Dataset: We apply our method on the MVTec AD dataset [7], which is a very challenging and comprehensive dataset for anomaly detection in the industrial inspection images. This dataset contains five texture categories (i.e., carpet, grid, leather, tile, and wood) images and ten object categories (i.e., bottle, cable, capsule, hazelnut, metal nut, pill, screw, toothbrush, transistor, and zipper) images. As there is no annotation of semantic structure for these 15 categories of images, only the low-level structure is used in our method. Therefore, only the image error $\mathcal{A}_{\text{img}}$ shown in (14) is used for AUC computation.

2) Quantitative Performance Evaluation: We compare our method with AE with L2 loss and SSIM loss [7], AnoGAN [8], Texture Inspection (TI) [71], CNN Feature Dictionary (CFD) [72], Cycle-GAN [73], Deep SVDD [28], VAE-GAN [9], GANomaly [38], and P-Net [20]. The results of AE (SSIM), AE (L2), AnoGAN [8], CFD [72], and TI [71] are adopted from [7] directly.

Besides the AUC that evaluates the image-level anomaly detection performance, we also report the region overlap between the predicted anomaly region and the ground truth as an evaluation metric to evaluate the anomaly segmentation (pixel-level anomaly detection) performance of our model following [7]. We segment a series of predicted anomaly regions by comparing the error between the normal images and the corresponding reconstructed images with an increased threshold. The searching of the threshold is not stopped until the area of the predicted anomaly region is just lower than the minimum defect area, which is predefined as a constant, and the threshold of the stopping point is utilized for segmenting the anomaly region in the test phase. The results are reported in Table I. We can see that our method achieves the best performance in terms of the average AUC and average anomaly region overlap on the average of all categories, which validates that our proposed MemSTC-Net is effective on industrial inspection images.

3) Qualitative Results: In this section, we analyze the qualitative reconstruction performance on different categories of images in the MVTec AD dataset. As shown in Figs. 5 and 6, we can observe that the abnormal region cannot be reconstructed reasonably. For example, the abnormal regions on the hazelnut are poorly reconstructed. We can also find that the high-frequency signal in the normal region is also distorted in the reconstructed images, e.g., metal conductors in the cable and the "rough surface" of the pill. This phenomenon is more obvious in the texture images (see Fig. 6) than in the object images (see Fig. 5), as the surface of the texture images is pretty rough. More high-frequency signals make the reconstruction error higher in the normal regions for the

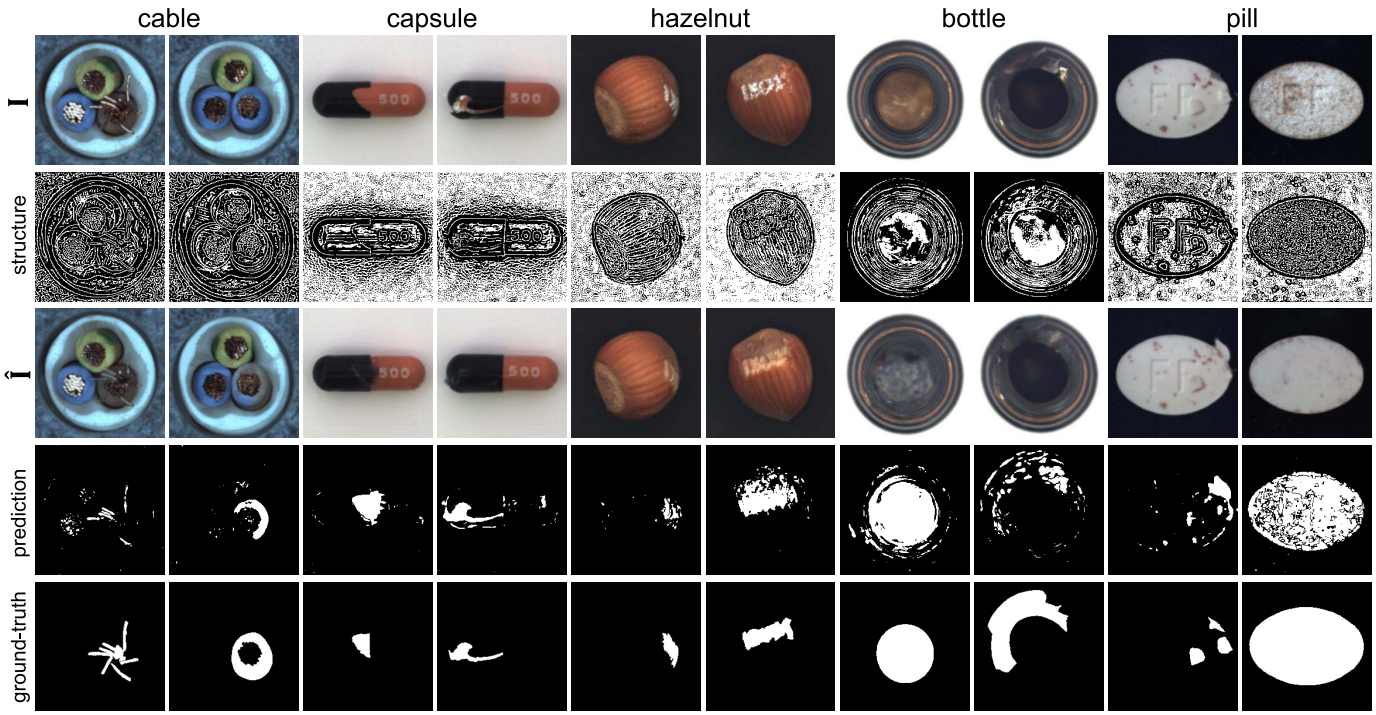

Fig. 5. Qualitative results of different object categories of images. **I** and $\hat{\mathbf{I}}$ denote original image and reconstructed image, respectively. "Structure" denotes the low-level structure (i.e., the Canny edge). "Ground truth" denotes the pixel-level abnormal annotation in MVTec AD dataset, and "prediction" is the pixel-level abnormal region predictions by the proposed method.

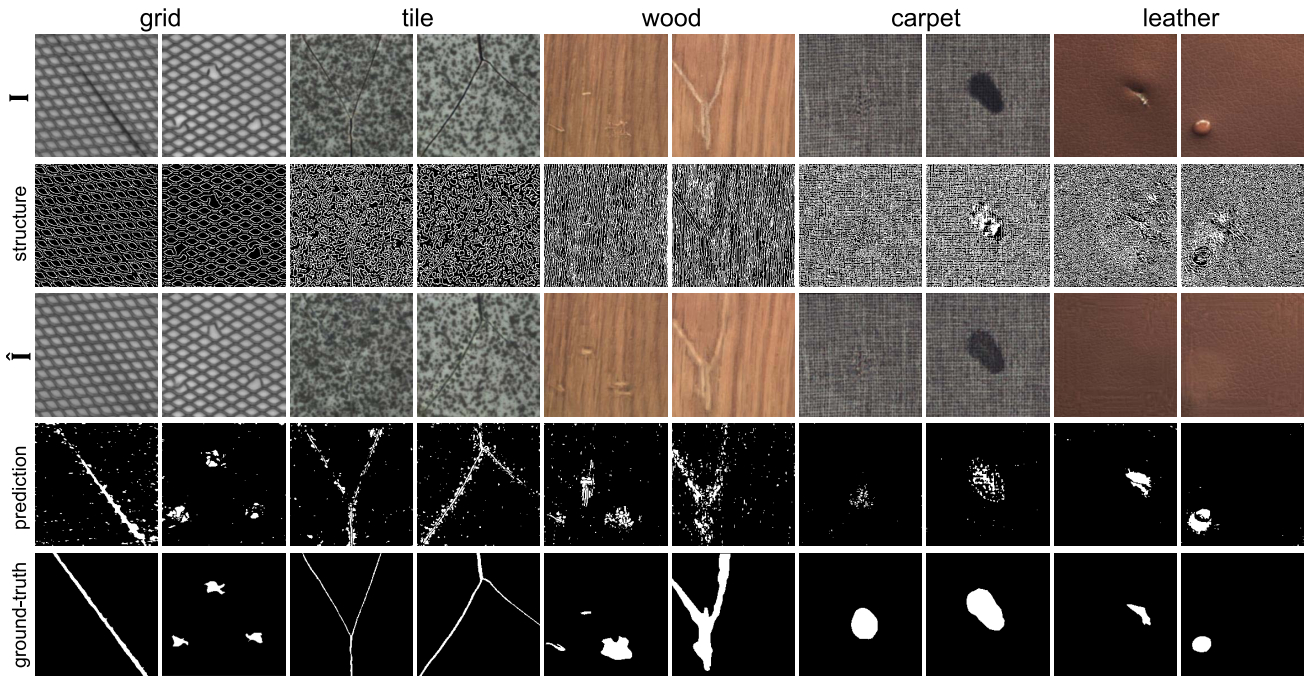

Fig. 6. Qualitative results of different texture categories of images.

texture images, such as the carpet and grid. Thus, more normal regions are segmented as abnormal regions in the texture images. As a result, the performance of our method in the texture images is relatively worse than in the object images.

C. Anomaly Detection on the Medical Images

1) Datasets: The datasets used in previous retinal image anomaly detection work [8], [37], [47] are not released; therefore, we evaluate our proposed method with a local hospital dataset [57] and a publicly available dataset [58].

a) Fundus multidisease diagnosis dataset (iSee) [57]: Previous retinal fundus datasets usually contain only one or two types of disease [74], [75], but, in the clinical diagnosis, many eye diseases can be observed in the retinal fundus image. Therefore, we use the iSee dataset [57] that contains multiple eye diseases to evaluate the sensitivity of the

TABLE II
PERFORMANCE COMPARISON ON DIFFERENT DATASETS

Method	RESC (OCT)	iSee (fundus)
Deep SVDD [28]	0.7440	0.6059
Auto-Encoder [11]	0.8207	0.6127
AnoGAN [8]	0.8481	0.6325
VAE-GAN [9]	0.9064	0.6969
Pix2Pix [66]	0.7934	0.6722
GANomaly [38]	0.9196	0.7015
Cycle-GAN [73]	0.8739	0.6699
geometric transformations [55]	0.8806	0.6384
P-Net [20]	0.9288	0.7245
MemSTC-Net	**0.9385**	**0.7568**

proposed method on multiple diseases. This dataset comprises of 10 000 retinal fundus images, including diabetic retinopathy (DR), age-related macular degeneration (AMD), glaucoma, pathological myopia (PM), and some other types of eye diseases. To evaluate the effectiveness of MemSTC-Net for image anomaly detection on this dataset, we use 4000 normal images as the training set, and we use the remaining 3000 normal images and 3000 abnormal images as our test set. Among the abnormal images in the test set, 480 images are with DR, 700 images are with AMD, 420 images are with glaucoma, 800 images are with PM, and 600 images are with other types of eye diseases.

b) Retinal edema segmentation challenge (RESC) dataset [58]: Retinal edema is a retinal disease, which can cause blurry vision and greatly affect the patient's life quality. As OCT images can be used to assist clinicians in diagnosing retinal edema, we utilize the RESC dataset, which is an OCT-based retinal edema segmentation dataset, for evaluation. Following the standard training/validating split of the dataset, we use the normal images in the original training set to train the model and use all test images for performance evaluation.

2) Performance Evaluation: We compare our method with VAE-GAN [9] proposed for Brain MRI images, AnoGAN [10] proposed for retinal OCT images, GANomaly [38] for X-ray security images, and AE-based anomaly detection [11]. We also compare MemSTC-Net with image-to-image translation networks, including Pix2Pix [66] and Cycle-GAN [73]. For Pix2Pix [66] and Cycle-GAN [73], we use the original image and structures extracted with the DA method to train the network and use the same measurement of anomaly score as ours for anomaly detection. Besides, as the use of cross-modality is a type of self-supervision, we compare MemSTC-Net with the well-known self-supervised image anomaly detection method, which uses geometric transformations [55]. We use the official implementation[1] provided in [55]. As reported in Table II, the proposed MemSTC-Net outperforms all baseline methods on the two medical datasets.

We further compute the average anomaly score (mean $\pm$ standard deviation) for both the normal images and the abnormal images on the fundus dataset with (16). The gap of these two scores is also calculated to measure the ability of our method and other baselines to discriminate the normal and abnormal images. A larger gap means that the normal and abnormal images can be more easily separated. The results reported in Table III show that our method achieves a larger gap than other baselines, which validates the effectiveness of our method for anomaly detection. We notice that the gap of AE is smaller than the gap of other baseline methods and our proposed method, which validates the assumption that the AE has a relatively low reconstruction error on abnormal images, and consequently, they are less sensitive to anomalies.

[1]https://github.com/izikgo/AnomalyDetectionTransformations

TABLE III
AVERAGE ANOMALY SCORE FOR THE NORMAL IMAGES, ABNORMAL IMAGES, AND THE GAP BETWEEN THESE TWO SCORES ON THE iSEE DATASET

Method	Anomaly Score $\mathcal{A}$		
	normal	abnormal	gap
Auto-Encoder [11]	0.737 ± 0.092	0.872 ± 0.128	0.135
VAE-GAN [9]	0.651 ± 0.077	0.842 ± 0.086	0.191
GANomaly [38]	0.717 ± 0.082	0.913 ± 0.063	0.196
P-Net [20]	0.823 ± 0.073	1.047 ± 0.060	0.224
MemSTC-Net	0.767 ± 0.049	1.025 ± 0.042	**0.258**

TABLE IV
AUC RESULTS OF SUBCLASS ON THE iSEE DATASET

Method	AMD	PM	Glaucoma	DR	Other
Auto-Encoder [11]	0.5463	0.7479	0.5604	0.6002	0.5479
AnoGAN [8]	0.5630	0.7499	0.5731	0.5704	0.6412
VAE-GAN [9]	0.5593	0.8412	0.6149	0.6590	0.7961
GANomaly [38]	0.5713	0.8336	0.6056	0.6627	0.8013
P-Net [20]	0.5688	**0.8726**	0.6103	**0.6830**	0.8069
MemSTC-Net	**0.6382**	0.8256	**0.6532**	0.6196	**0.9223**

We also report the AUC results of our method for the five subclasses in the iSee dataset (i.e., AMD, PM, glaucoma, DR, and other disease classes) in Table IV. We can observe that the proposed MemSTC-Net outperforms P-Net on three sub-classes (i.e., AMD, glaucoma, and other types of disease).

3) Comparison Between MemSTC-Net and P-Net: The P-Net [20] reconstructs the image from the fusion of the semantic structure feature and the texture feature. The texture feature is extracted from the original image via a texture AE. However, since the input of the texture encoder is the original image, the texture encoder probably introduces abnormal information for abnormal image reconstruction in the test phase, which is unfavorable for anomaly detection. Thus, in this article, the proposed MemSTC-Net removes the texture encoder in P-Net and introduces an STCM module between the semantic structure and the reconstructed image to memorize the structure-texture correspondence for the normal images. To investigate the effectiveness of the proposed STCM module, we conduct quantitative and qualitative experiments that are shown in Table V and Fig. 7, respectively.

First, we show that both image and semantic structure are necessary for anomaly detection. As reported in Table V, the result of P-Net w/o SR (i.e., encoding the structure-texture relation with P-Net) is better than that with the single input (i.e., only texture or semantic structure feature for image reconstruction)-based reconstruction. In particular, the results of P-Net w/o SR on the iSee dataset arise 7% and 3% on AUC compared with the results of a single feature for reconstruction.

Then, we show the effectiveness of the STCM module, which captures the structure-texture correspondence by a memory. Note that, to fairly compare with P-Net w/o SR, the MemSTC-Net w/o SR (i.e., encoding structure-texture correspondence with STCM module) here does not use the low-level structure. As reported in Table V, the result of MemSTC-Net w/o SR is better than that with P-Net w/o SR.

The qualitative comparison also validates the effectiveness of the proposed STCM module. From Fig. 7, we can observe the following.

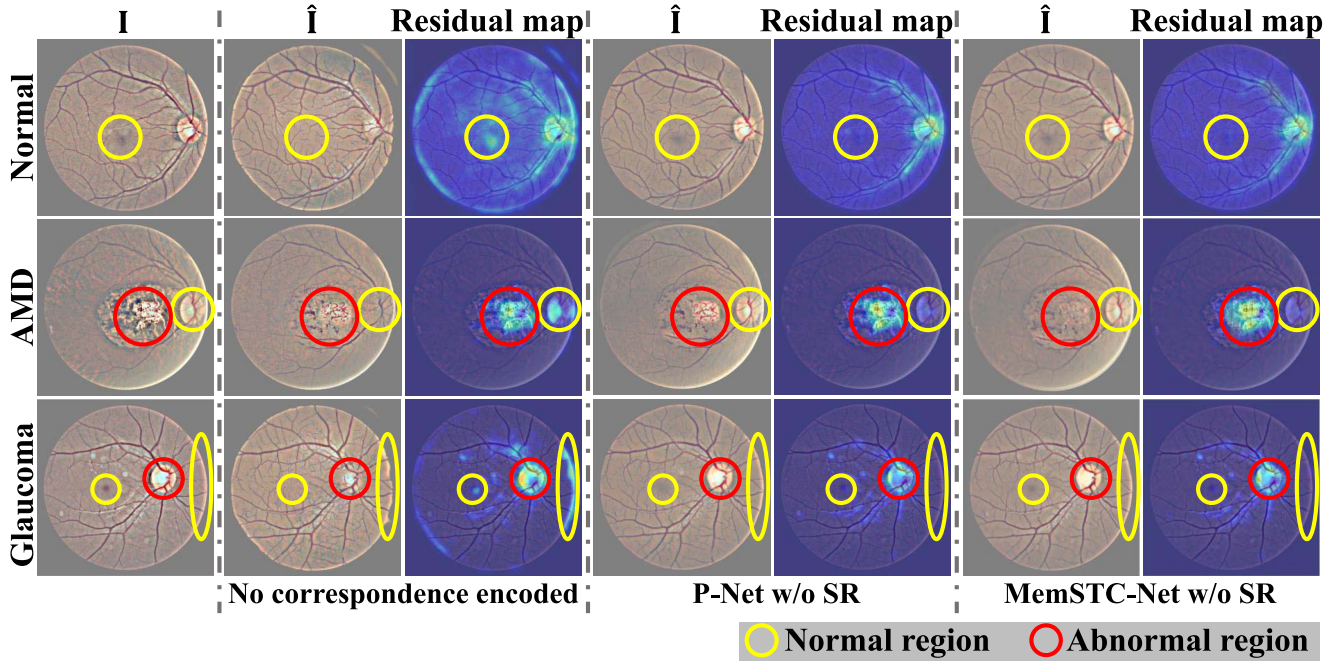

Fig. 7. Qualitative comparison of the ways to encode structure-texture correspondence. The residual map is fused by the original image **I** and the absolute difference between the original image and its reconstructed image $\hat{\mathbf{I}}$. "No correspondence encoded" denotes only taking semantic structure as input; in this way, there is no structure-texture correspondence encoded. The P-Net w/o SR means that encoding structure-texture relation with P-Net, and the MemSTC-Net w/o SR means that encoding structure-texture correspondence with the proposed STCM module.

TABLE V
COMPARISON OF THE WAYS TO ENCODE STRUCTURE-TEXTURE CORRESPONDENCE

Method	RESC	iSee
texture feature [20]	0.8219	0.6487
semantic structure feature [20]	0.8277	0.6914
P-Net w/o SR [20]	0.8518	0.7196
MemSTC-Net w/o SR	**0.8846**	**0.7367**

1) When only taking the semantic structure as input (i.e., there is no structure-texture correspondence encoded), the texture (e.g., the area of the macular and optic disk) cannot be well reconstructed. This is because of the lack of texture information.
2) In the results of P-Net w/o SR, the reconstruction of the normal region is better than that without structure-texture correspondence encoded. As illustrated in the yellow circles, the reconstruction result of normal regions (e.g., macular and optic disk area) of P-Net w/o SR is better.
3) Compared with P-Net w/o SR, the reconstruction of the abnormal region (circled with red color) with MemSTC-Net w/o SR is more like the normal. This enlarges the reconstruction error for the abnormal images and boosts the performance of anomaly detection. These qualitative results validate the effectiveness of replacing the texture encoder with the proposed STCM module to encode the structure-texture correspondence.

4) Ablation Study: In the previous sections, comprehensive comparisons between the previous P-Net and the proposed MemSTC-Net have proved the effectiveness of the major components of the proposed method. This section conducts several ablation studies on the iSee and RESC datasets to explore other different components in detail.

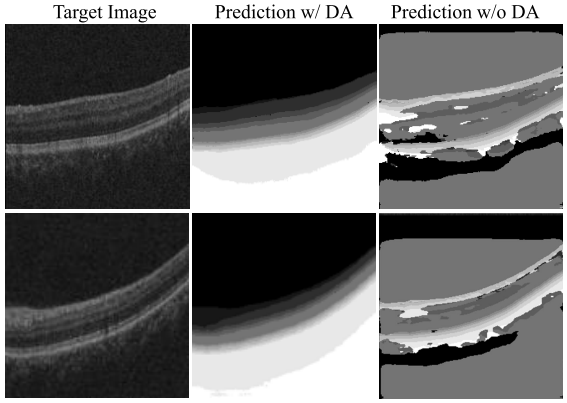

Fig. 8. Qualitative results of DA for OCT images. The structure of target image is well extracted well with DA.

a) Domain adaptation: Since there exists domain discrepancy between the source images and the target images, if we train a semantic structure extraction model without DA, the quality of the semantic structure (third column in Fig. 8) is terrible for image reconstruction. The quantitative results also validate the necessity of DA. As reported in Table VI, by comparing the results in row 1 and row 2, it can be observed that the performance without using DA decreases enormously, i.e., 24% AUC and 16% AUC on both datasets.

b) Effectiveness of using two types of structure: Our MemSTC-Net leverages both semantic structure and low-level structure to reconstruct the image. To study the effectiveness of this design, we conduct quantitative experiments, and the results are reported in Table VI. By comparing the results

TABLE VI
ABLATION STUDIES OF DA AND STRUCTURE SELECTION

	Semantic Structure	with DA	Low-level Structure	AUC	
				RESC	iSee
1	✓	×	×	0.6480	0.5791
2	✓	✓	×	0.8846	0.7367
3	×	-	✓	0.8126	0.6965
4	✓	✓	✓	**0.9018**	**0.7488**

TABLE VII
RESULTS WITH AND WITHOUT THE STCM MODULE

	$\boldsymbol{M}_s$	$\boldsymbol{M}_e$	RESC	iSee
1	Concat	Concat	0.8368	0.6671
2	STCM	Concat	0.8984	0.7291
3	Concat	STCM	0.8455	0.6893
4	STCM	STCM	**0.9018**	**0.7488**

TABLE VIII
STUDY OF THE CONCATENATING MANNERS IN THE MEMORY

	key	value	RESC	iSee
1	$z_s \oplus z_t$	z_t	0.8963	0.7403
2	z_s	$z_s \oplus z_t$	0.8949	0.7412
3	$z_s \oplus z_t$	$z_s \oplus z_t$	0.9007	0.7361
4	z_s	z_t	**0.9018**	**0.7488**

(row 2 versus row 4, and row 3 versus row 4), it can be validated that using the semantic structure and low-level structure cooperatively is efficient. Particularly, although the performance of only using low-level structure is lower than only using semantic structure, combining the low-level structure and the semantic structure together boosts the performance.

c) Improvement with STCM module: The proposed STCM module memorizes the structure-texture correspondence by storing the key–value pairs, where the structure feature is the key and the texture feature is the value. To validate the effectiveness of this design, we compare the STCM module with a simple baseline without the memory module. In this baseline (denoted as "Concat"), the structure feature and the retrieved texture feature are concatenated as one entry that is input to the following decoder. As reported in Table VII, it can be observed that: 1) using the STCM module for both semantic structure and low-level structure improve the performance and 2) memorizing semantic structure-texture correspondence is more effective than memorizing low-level structure-texture correspondence.

Besides using z_s as key and z_t as value, there are three other cases concatenating z_s and z_t as one item in the memory: 1) concatenating z_s and z_t (denoted as $z_s \oplus z_t$) as the key; 2) concatenating z_s and z_t as the value; and 3) concatenating z_s and z_t both as the key and the value. The results are reported in Table VIII, and these other cases do not bring improvement.

d) Improvements with attention-guided fusion module: In our proposed method, rather than simply averaging the reconstructed images with the two types of structure, we employ an attention block to learn a weight to fuse the two reconstructed images automatically. The results are reported in Table IX; the proposed fusion strategy with the attention outperforms the simply averaging by 2.9% AUC and 0.9% AUC on RESC and iSee datasets, respectively.

e) Improvements with SR module: The SR module contains two structure consistency losses, i.e., $\mathcal{L}_{\text{sr}}^{s}$ constraining the consistency between $\mathbf{S}$ and $\hat{\mathbf{S}}$, and $\mathcal{L}_{\text{sr}}^{e}$ constraining the consistency between $\mathbf{E}$ and $\hat{\mathbf{E}}$. The SR module behaves like a regularizer to enforce the consistency between the input image and the reconstructed image. We report the results of P-Net [20] and our MemSTC-Net trained with and without SR module. As reported in Table X, the performance of MemSTC-Net is improved with the SR module, which verifies the effectiveness of enforcing the consistency between the input image and the reconstruction for image anomaly detection. In addition, the performance improvements on REST are large than iSee, which shows that SR is more effective on the OCT modality than that on the fundus modality.

TABLE IX
COMPARISON OF DIFFERENT FUSION STRATEGIES OF TWO RECONSTRUCTED IMAGES

	$\hat{\mathbf{I}}_s$	$\hat{\mathbf{I}}_e$	$(\hat{\mathbf{I}}_s + \hat{\mathbf{I}}_e)/2$	proposed
RESC	0.8846	0.8126	0.8733	**0.9018**
iSee	0.7367	0.6965	0.7394	**0.7488**

TABLE X
RESULTS OF USING THE SR MODULE

Method	RESC	iSee
MemSTC-Net w/o SR	0.9018	0.7488
MemSTC-Net	**0.9385**(Δ=+3.7%)	**0.7568**(Δ=+0.8%)

5) Hyperparameters Analysis: In this section, we conduct several experiments on the iSee and RESC datasets to analyze the hyperparameters during the training and test.

a) Analysis of the hyperparameters in objective function: During the optimization, we use the objective function in (13) to train the proposed model. It is essential to balance the weights between reconstruction loss ($\mathcal{L}_{\text{rec}}$), adversarial loss ($\mathcal{L}_{\text{adv}}$), and the SR items ($\mathcal{L}_{\text{sr}}^{s}$ and $\mathcal{L}_{\text{sr}}^{e}$). To explore the influence of these hyperparameters (λ_{rec}, λ_{adv}, λ_{sr}^{s}, and λ_{sr}^{e}), we conduct following experiments.

1) Fix $\lambda_{\text{adv}} = 0.5, \lambda_{\text{sr}}^{s} = 0.05$, and $\lambda_{\text{sr}}^{e} = 0.01$, and change λ_{rec}.
2) Fix $\lambda_{\text{rec}} = 1, \lambda_{\text{sr}}^{s} = 0.05$, and $\lambda_{\text{sr}}^{e} = 0.01$, and change λ_{adv}.
3) Fix $\lambda_{\text{rec}} = 1, \lambda_{\text{adv}} = 0.5$, and $\lambda_{\text{sr}}^{e} = 0.01$, and change λ_{sr}^{s}.
4) Fix $\lambda_{\text{rec}} = 1, \lambda_{\text{adv}} = 0.5$, and $\lambda_{\text{sr}}^{s} = 0.05$, and change λ_{sr}^{e}.

Fig. 9(a) and (b) shows that a larger $\lambda_{\text{rec}}/\lambda_{\text{adv}}$ does not always improve AUC. As shown in Fig. 9(c) and (d), the effect of two SR items is different on different datasets. For example, the effect of λ_{sr}^{s} on the iSee dataset is more robust than that on the RESC dataset. On the contrary, the effect of λ_{sr}^{e} on the RESC dataset is more robust than that on the iSee dataset. Empirically, we set $\lambda_{\text{rec}} = 1, \lambda_{\text{adv}} = 0.1, \lambda_{\text{sr}}^{s} = 0.05$, and $\lambda_{\text{sr}}^{e} = 0.01$ on all datasets in our experiments.

b) Evaluation of the memory size (k): Besides the hyperparameters in objective functions, the memory size is also important for our model. We gradually change the memory size in {512, 1024, 2048, 4096}, and the results are shown in Fig. 10. It can be found that: 1) when $k = 2048$, the model achieves the best performance on both RESC (OCT) and iSee (fundus) datasets; therefore, we set $k = 2048$ in all experiments and 2) when $k > 2048$ or $k < 2048$, the performance descends. On the one hand, if the memory size is too small, the STCM module is incapable to memorize the structure-texture correspondence in normal images, leading to

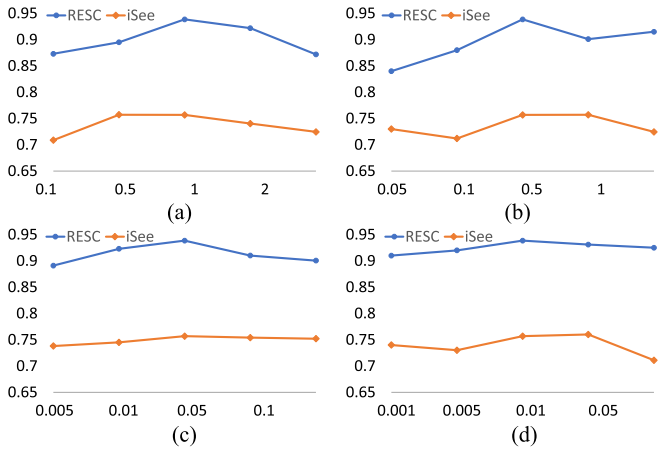

Fig. 9. AUC results versus different weights of the hyperparameters. (a) and (b) show that a larger $\lambda_{\text{rec}}/\lambda_{\text{adv}}$ do not always improve AUC. (c) and (d) show that the effect of two SR items is different on different datasets. (a) AUC versus λ_{rec}. (b) AUC versus λ_{adv}. (c) AUC versus λ_{sr}^{s}. (d) AUC versus λ_{sr}^{e}.

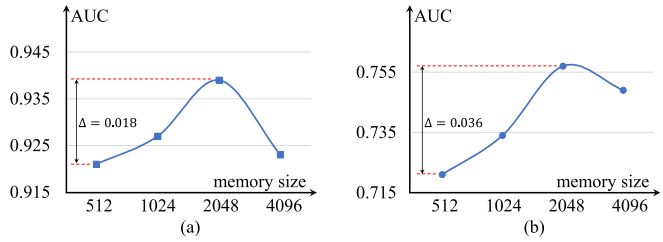

Fig. 10. AUC results on (a) RESC (OCT) and (b) iSee (fundus) with different memory sizes. When $k = 2048$, the model achieves the best performance on both RESC and iSee datasets.

TABLE XI
RESULTS OF DIFFERENT γ'S ON RESC AND ISEE DATASETS

γ	0.0	0.2	0.4	0.6	0.8	1.0
RESC	0.8507	0.9120	0.9241	0.9305	**0.9385**	0.9137
iSee	0.7022	0.7227	**0.7568**	0.7436	0.7271	0.7108

the poor reconstruction of the normal images. On the other hand, if the memory size is too big, the capacity of the model becomes high, leading to a good reconstruction of the abnormal images.

In addition, the fluctuation of AUC results on the iSee dataset is larger than that on the RESC dataset. This is because the pattern in the fundus image is more complex than the pattern in the OCT dataset, leading to that the model is more sensitive to the fundus than that to OCT.

c) Evaluation of γ: In the test phase, we use (16) to measure the anomaly score. $\gamma = 0$ denotes that only image error $\mathcal{A}_{\text{img}} = \|\mathbf{I}-\hat{\mathbf{I}}\|_1$ is used for anomaly detection, and $\gamma = 1$ means that only semantic structure error $\mathcal{A}_{\text{struct}} = \|\mathbf{S}-\hat{\mathbf{S}}\|_1$ is used for anomaly detection. We vary γ and show the results in Table XI. We can see that the performance of anomaly detection using only $\mathcal{A}_{\text{img}}$ is worse than using only $\mathcal{A}_{\text{struct}}$. The possible reason is that the semantic structure is more evident than image for anomaly detection, which agrees with the practice of clinicians. Combining $\mathcal{A}_{\text{img}}$ and $\mathcal{A}_{\text{struct}}$ together leads to a better performance. Our proposed method achieves the best performance when $\gamma = 0.8$ on RESC. For the iSee dataset, the best performance is obtained when $\gamma = 0.4$.

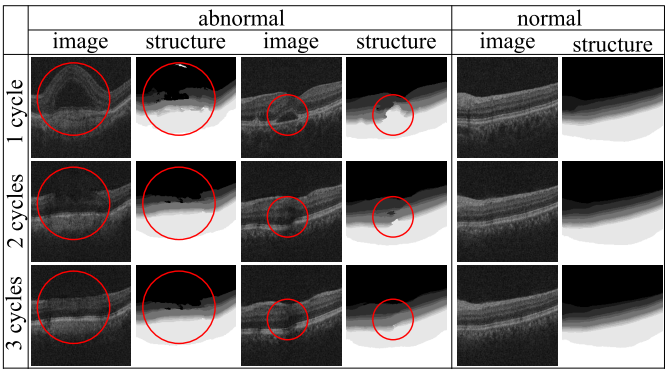

Fig. 11. Qualitative results with different numbers of cycles. It can be observed that the texture and the layerwise structure between the original and reconstructed ones in the normal sample are consistent, while the consistency in abnormal samples is broken. The anomalies are denoted by red circles.

TABLE XII
RESULTS OF DIFFERENT NUMBERS OF CYCLES FOR TEST ON THE RESC DATASET

Cycle number in test	1	2	3
AUC	0.9385	0.9389	0.9391

6) Multiple Cycles in MemSTC-Net: We can find that the reconstruction results for the abnormal images are worse than the normal images, and this is why we can detect the anomaly. This motivates us to feed the reconstructed image as the input to the network again to get a reconstructed image, and this process can be conducted for multiple cycles. Therefore, the reconstruction result for the abnormal images becomes worse and worse, which facilitates anomaly detection. In Fig. 11, we show the qualitative effect of more cycles in test phase, and we report the quantitative results in Table XII. In the multiple cycles in the test phase, the abnormal lesion becomes more and more similar to normal patterns, which is a little like "anomalies repairing." Such a phenomenon is more obvious in the semantic structure map. For the normal image, both the image and structure map remain the same even after multiple cycles. Thus, more cycles would enlarge the reconstruction error for abnormal images and retain the same reconstruction error for normal ones, which explains the phenomenon that more cycles in the test phase improve the anomaly detection. Except for this section, we only conduct one cycle in the test for a fair comparison with other baselines.

D. Unseen Disease Discovery on the Retinal Fundus Images

In the real clinical scenario, clinicians can recognize the images with diseases that they have never seen before. It is desirable to achieve this capability in an intelligent diagnosis system. This task is termed unseen disease discovery [76], where the test set contains the images of disease categories not appeared in the training set.

To evaluate the performance of our method under such a setting, we use 4000 normal images and 350 images with AMD, 400 images with PM, 210 images with glaucoma, and 240 images with DR in our iSee dataset as the training set. We use the remaining images (i.e., 600 images with other types of eye disease) for unseen disease discovery in the test phase. We define normal, AMD, PM, glaucoma, and DR as the seen classes and the other diseases as the unseen classes.

TABLE XIII
AUC RESULTS OF UNSEEN DISEASE DISCOVERY ON THE ISEE DATASET

Method	AUC
Deep SVDD [28]	0.5794
Auto-Encoder [11]	0.5244
Pix2Pix [66]	0.5249
VAE-GAN [9]	0.6172
GANomaly [38]	0.6240
P-Net [20]	0.6593
Our Method	**0.6622**

We report the AUC performance of different methods under such setting in Table XIII. We can see that our method outperforms other methods. The reason for the success of our method is that our network can memorize the correspondence between the structure and the texture. For the seen images with certain diseases in the training set, the network can utilize the structure-texture correspondence for reconstruction. However, the memorized structure-texture correspondence of the seen diseases in the training phase cannot generalize to the unseen diseases in the test phase, leading to a large reconstruction error, which can be used for unseen disease discovery.

V. CONCLUSION

In this work, we propose a novel MemSTC-Net for image anomaly detection. The motivation of our method is that the correlation between the structure and texture in normal images is stronger than that in abnormal images; thus, the normal texture can be inferred from the normal structure, while it is hard to infer the abnormal texture from the abnormal structure. Based on this observation, we reconstruct the image from the structure-texture correspondence that is stored in the proposed STCM module. Besides, we extract two types of structures (i.e., the semantic structure and the low-level structure) from the original images. The reconstructions from these two types of structures are fused together to get the final reconstructed image in the attention-guided fusion module. Finally, we extract the structures from the reconstructed images and minimizing the difference between the structures extracted from the original image and that from the reconstructed image in training. In the test, we combine the image reconstruction error and semantic structure error as a measurement for image anomaly detection. Extensive experiments validate the effectiveness of our approach.

ACKNOWLEDGMENT

The authors would like to thank the Editor-in-Chief, the Associate Editor, and all anonymous reviewers for their valuable and constructive comments.

REFERENCES

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surveys*, vol. 41, no. 3, pp. 1–58, 2009.
[2] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," 2019, *arXiv:1901.03407*. [Online]. Available: http://arxiv.org/abs/1901.03407
[3] G. Pang, C. Shen, L. Cao, and A. Hengel, "Deep learning for anomaly detection: A review," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–38, 2020.
[4] K. Zhou *et al.*, "Multi-cell multi-task convolutional neural networks for diabetic retinopathy grading," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 2724–2727.
[5] Z. Tu *et al.*, "SUNet: A lesion regularized model for simultaneous diabetic retinopathy and diabetic macular edema grading," in *Proc. IEEE 17th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2020, pp. 1378–1382.
[6] X. Li, X. Hu, L. Yu, L. Zhu, C.-W. Fu, and P.-A. Heng, "CANet: Cross-disease attention network for joint diabetic retinopathy and diabetic macular edema grading," *IEEE Trans. Med. Imag.*, vol. 39, no. 5, pp. 1483–1493, Nov. 2019.
[7] 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. 9592–9600.
[8] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Proc. Int. Conf. Inf. Process. Med. Imag.* Cham, Switzerland: Springer, 2017, pp. 146–157.
[9] 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, 2018, pp. 161–169.
[10] X. Chen and E. Konukoglu, "Unsupervised detection of lesions in brain MRI using constrained adversarial auto-encoders," in *Proc. Med. Imag. Deep Learn. (MIDL)*, 2018.
[11] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 665–674.
[12] D. Zimmerer, S. Kohl, J. Petersen, F. Isensee, and K. Maier-Hein, "Context-encoding variational autoencoder for unsupervised anomaly detection," in *Proc. Int. Conf. Med. Imag. Deep Learn. (MIDL)*, 2019.
[13] K. Zhou *et al.*, "Sparse-GAN: Sparsity-constrained generative adversarial network for anomaly detection in retinal OCT image," in *Proc. IEEE 17th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2020, pp. 1227–1231.
[14] D. Gong *et al.*, "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.
[15] B. Zong *et al.*, "Deep autoencoding Gaussian mixture model for unsupervised anomaly detection," in *Proc. Int. Conf. Learn. Represent.*, 2018.
[16] K. Boyd. (2019). *What is Diabetic Retinopathy*. [Online]. Available: https://www.aao.org/eye-health/diseases/what-is-diabetic-retinopathy
[17] C. A. Puliafito *et al.*, "Imaging of macular diseases with optical coherence tomography," *Ophthalmology*, vol. 102, no. 2, pp. 217–229, 1995.
[18] S. J. Zinreich *et al.*, "Fungal sinusitis: Diagnosis with CT and MR imaging," *Radiology*, vol. 169, no. 2, pp. 439–444, Nov. 1988.
[19] M. E. Hartnett, J. J. Weiter, G. Staurenghi, and A. E. Elsner, "Deep retinal vascular anomalous complexes in advanced age-related macular degeneration," *Ophthalmology*, vol. 103, no. 12, pp. 2042–2053, Dec. 1996.
[20] K. Zhou *et al.*, "Encoding structure-texture relation with P-Net for anomaly detection in retinal images," in *Proc. ECCV*, 2020, pp. 360–377.
[21] *Novelty Outlier Detection*. [Online]. Available: https://scikit-learn.org/stable/modules/outlier_detection.html
[22] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.
[23] A. Boukerche, L. Zheng, and O. Alfandi, "Outlier detection: Methods, models, and classification," *ACM Comput. Surveys*, vol. 53, no. 3, pp. 1–37, 2020.
[24] Y. Xia, X. Cao, F. Wen, G. Hua, and J. Sun, "Learning discriminative reconstructions for unsupervised outlier removal," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1511–1519.
[25] C. C. Aggarwal, "Outlier analysis," in *Data Mining*. Springer, 2015, pp. 237–263.
[26] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 582–588.
[27] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, Sep. 1962.
[28] L. Ruff *et al.*, "Deep one-class classification," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4393–4402.
[29] L. Li, J. Hansman, R. Palacios, and R. Welsch, "Anomaly detection via a Gaussian mixture model for flight operation and safety monitoring," *Transp. Res. C, Emerg. Technol.*, vol. 64, pp. 45–57, Mar. 2016.
[30] P. D. McNicholas and T. B. Murphy, "Parsimonious Gaussian mixture models," *Statist. Comput.*, vol. 18, no. 3, pp. 285–296, Sep. 2008.

[31] D. J. Miller, G. Kesidis, and Z. Qiu, "Unsupervised parsimonious cluster-based anomaly detection (PCAD)," in *Proc. IEEE 28th Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2018, pp. 1–6.

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

[33] W. Yu, W. Cheng, C. C. Aggarwal, K. Zhang, H. Chen, and W. Wang, "NetWalk: A flexible deep embedding approach for anomaly detection in dynamic networks," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 2672–2681.

[34] G. Pang, L. Cao, L. Chen, and H. Liu, "Learning representations of ultrahigh-dimensional data for random distance-based outlier detection," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 2041–2050.

[35] D. Carrera, G. Boracchi, A. Foi, and B. Wohlberg, "Detecting anomalous structures by convolutional sparse models," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2015, pp. 1–8.

[36] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 2014, pp. 2672–2680.

[37] T. Schlegl, P. Seeböck, S. M. Waldstein, G. Langs, and U. Schmidt-Erfurth, "f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks," *Med. Image Anal.*, vol. 54, pp. 30–44, May 2019.

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

[39] K. Yamanishi, J.-I. Takeuchi, G. Williams, and P. Milne, "On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms," *Data Mining Knowl. Discovery*, vol. 8, no. 3, pp. 275–300, 2004.

[40] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 93–104.

[41] H. Xu, C. Caramanis, and S. Mannor, "Outlier-robust PCA: The high-dimensional case," *IEEE Trans. Inf. Theory*, vol. 59, no. 1, pp. 546–572, Jan. 2013.

[42] S. Wang *et al.*, "Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network," in *Proc. NeurIPS*, 2019, pp. 5960–5973.

[43] H. Tang, X. Qi, D. Xu, P. H. S. Torr, and N. Sebe, "Edge guided GANs with semantic preserving for semantic image synthesis," 2020, *arXiv:2003.13898*. [Online]. Available: http://arxiv.org/abs/2003.13898

[44] Y. Ren, X. Yu, R. Zhang, T. H. Li, S. Liu, and G. Li, "StructureFlow: Image inpainting via structure-aware appearance flow," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 181–190.

[45] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi, "EdgeConnect: Structure guided image inpainting using edge prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019.

[46] C. Wang, X. Chen, S. Min, Z.-J. Zha, and J. Wang, "Structure-guided deep video inpainting," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Oct. 28, 2020, doi: 10.1109/TCSVT.2020.3034422.

[47] C. Zhang *et al.*, "Memory-augmented anomaly generative adversarial network for retinal OCT images screening," in *Proc. IEEE 17th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2020, pp. 1971–1974.

[48] H. Park, J. Noh, and B. Ham, "Learning memory-guided normality for anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14372–14381.

[49] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, May 4, 2020, doi: 10.1109/TPAMI.2020.2992393.

[50] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. Int. Conf. Learn. Represent.*, 2018.

[51] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 69–84.

[52] R. S. Cruz, B. Fernando, A. Cherian, and S. Gould, "Visual permutation learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 12, pp. 3100–3114, Dec. 2019.

[53] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 649–666.

[54] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2536–2544.

[55] I. Golan and R. El-Yaniv, "Deep anomaly detection using geometric transformations," in *Proc. NeurIPS*, 2018.

[56] Z. Li *et al.*, "Superpixel masking and inpainting for self-supervised anomaly detection," in *Proc. 31st Brit. Mach. Vis. Conf.*, 2020, pp. 7–10.

[57] Y. Yan *et al.*, "Oversampling for imbalanced data via optimal transport," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 5605–5612.

[58] J. Hu, Y. Chen, and Z. Yi, "Automated segmentation of macular edema in OCT using deep neural networks," *Med. Image Anal.*, vol. 55, pp. 216–227, Jul. 2019.

[59] A. D. Hoover, V. Kouznetsova, and M. Goldbaum, "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response," *IEEE Trans. Med. Imag.*, vol. 19, no. 3, pp. 203–210, Mar. 2000.

[60] J. Staal, M. D. Abràmoff, M. Niemeijer, M. A. Viergever, and B. van Ginneken, "Ridge-based vessel segmentation in color images of the retina," *IEEE Trans. Med. Imag.*, vol. 23, no. 4, pp. 501–509, Apr. 2004.

[61] J. Cheng *et al.*, "Speckle reduction in 3D optical coherence tomography of retina by A-scan reconstruction," *IEEE Trans. Med. Imag.*, vol. 35, no. 10, pp. 2270–2279, Oct. 2016.

[62] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker, "Learning to adapt structured output space for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7472–7481.

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

[64] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[65] D. Medhi and K. Ramasamy, *Network Routing: Algorithms, Protocols, and Architectures*. San Mateo, CA, USA: Morgan Kaufmann, 2017.

[66] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.

[67] Z. Chai *et al.*, "Perceptual-assisted adversarial adaptation for choroid segmentation in optical coherence tomography," in *Proc. IEEE 17th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2020, pp. 1966–1970.

[68] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Proc. Int. Conf. Learn. Represent.*, 2015.

[69] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.

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

[71] T. Böttger and M. Ulrich, "Real-time texture error detection on textured surfaces with compressed sensing," *Pattern Recognit. Image Anal.*, vol. 26, no. 1, pp. 88–94, Jan. 2016.

[72] P. Napoletano, F. Piccoli, and R. Schettini, "Anomaly detection in nanofibrous materials by CNN-based self-similarity," *Sensors*, vol. 18, no. 2, p. 209, Jan. 2018.

[73] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2223–2232.

[74] P. Porwal *et al.*, "IDRiD: Diabetic retinopathy–segmentation and grading challenge," *Med. Image Anal.*, vol. 59, Jan. 2019, Art. no. 101561.

[75] J. I. Orlando *et al.*, "REFUGE challenge: A unified framework for evaluating automated methods for glaucoma assessment from fundus photographs," *Med. Image Anal.*, vol. 59, Jan. 2020, Art. no. 101570.

[76] Y. Xiao *et al.*, "Open-set OCT image recognition with synthetic learning," in *Proc. IEEE 17th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2020, pp. 1788–1792.