UNIVERSITÀ
di VERONA
Dipartimento
di INFORMATICA

Feature Augmented Variational Autoencoder (FAVAE) for Anomaly Detection and Localization on MVTec Datasets

Presented by:

Abel Abebe
Kidus D. Bellete

Submitted to:

Professor Vittorio Murino
Dr. Andrea Avogaro

Academic Year 2023-2024

Contents

Chapter 1

Introduction

Anomaly detection in industrial settings is a critical task for maintaining product quality and reducing manufacturing costs. This project introduces the Feature Augmented Variational Autoencoder (FAVAE), a novel approach to unsupervised anomaly detection and localization in industrial images. The FAVAE combines the generative power of Variational Autoencoders (VAEs) with feature-level information from a pre-trained convolutional neural network to enhance detection and localization accuracy.

The FAVAE architecture leverages the strengths of both VAEs and pre-trained CNNs to create a robust anomaly detection system. By incorporating high-level feature representations from a pre-trained CNN, the FAVAE can capture complex spatial and semantic information that may be missed by traditional VAEs alone. This feature augmentation allows for more precise reconstruction of normal samples and better discrimination of anomalous regions. Our approach addresses the limitations of traditional anomaly detection methods by:

- Utilizing deep learning techniques to improve detection rates and reduce false positives
- Enhancing scalability across various industrial applications
- Leveraging knowledge distillation from large-scale image recognition tasks

The MVTec Anomaly Detection dataset serves as the testbed for our FAVAE model, providing a comprehensive collection of industrial images with various defects. This dataset encompasses a wide range of industrial objects and textures, presenting a challenging and realistic scenario for anomaly detection algorithms. By improving the accuracy and reliability of automated inspection systems, this project has the potential to enable more proactive quality control measures. This can lead to significant cost savings, improved product quality, and enhanced competitiveness in the global manufacturing market.

Chapter 2

Motivation and Rationale

The industrial sector faces constant challenges in maintaining product quality while minimizing costs. Automated visual inspection systems play a crucial role in this context, but they must be capable of detecting a wide range of potential defects, including subtle anomalies that may be difficult for human inspectors to consistently identify.

As production processes become more complex and output volumes increase, traditional quality control methods relying on human inspection are becoming inadequate. These manual processes are time-consuming, expensive, and prone to errors due to fatigue and inconsistency. Moreover, many modern manufacturing defects are too subtle or occur too quickly for human detection, potentially leading to product failures and customer dissatisfaction.

The Feature Augmented Variational Autoencoder (FAVAE) addresses the need for a more sophisticated and accurate anomaly detection system. By augmenting the Variational Autoencoder with feature-level information from a pre-trained convolutional neural network, we aim to create a model that can learn more robust representations of normal patterns in industrial images. This approach leverages transfer learning, utilizing knowledge from large-scale image recognition tasks to enhance anomaly detection capabilities.

The unsupervised nature of FAVAE makes it particularly suitable for industrial applications where labeled datasets of defects are often scarce. By improving the accuracy and reliability of automated inspection systems, this project has the potential to enable more proactive quality control measures, leading to cost savings, improved product quality, and enhanced competitiveness in the global market.

Chapter 3

State of the Art

Recent advancements in unsupervised anomaly detection have led to the development of various deep learning-based methods. These approaches aim to learn normal data patterns and identify deviations as anomalies. Some notable techniques include:

3.1 Autoencoders and Variants

Autoencoders (AEs) have been widely used for anomaly detection due to their ability to learn compact representations of input data. Variational Autoencoders (VAEs) [7] extend this concept by introducing a probabilistic encoder, allowing for better generalization and the generation of new samples. VAEs have shown promise in anomaly detection tasks, particularly for complex, high-dimensional data such as industrial images.

3.2 One-Class Classification Methods

Deep SVDD [2] adapts the traditional Support Vector Data Description to deep neural networks, learning a hypersphere that encloses normal data points while excluding anomalies. This method has demonstrated effectiveness in various anomaly detection scenarios.

3.3 Generative Adversarial Networks

AnoGAN [4] utilizes Generative Adversarial Networks (GANs) for anomaly detection. By learning to generate normal samples, AnoGAN can identify anomalies as instances that the generator struggles to reconstruct accurately.

3.4 Hybrid Approaches

DAGMM [3] combines dimensionality reduction with density estimation, using an autoencoder coupled with a Gaussian Mixture Model to detect anomalies in latent space.

While these methods have shown promising results, they often face challenges when dealing with the complexity and variability of real-world industrial images. VAEs, in particular, may struggle to capture fine-grained details necessary for detecting subtle anomalies.

Our proposed FAVAE method aims to address these limitations by enhancing the representational capacity of traditional VAEs. By optimizing the VAE architecture for industrial image analysis, we seek to improve anomaly detection and localization performance on the MVTec dataset.

Chapter 4

Objectives

4.1 General Objective

To develop and evaluate a Feature Augmented Variational Autoencoder (FAVAE) for improved anomaly detection and localization on the MVTec datasets.

4.2 Specific Objectives

1. Implement a FAVAE architecture that enhances the representational capacity of traditional VAEs.
2. Develop a training methodology that incorporates feature-level information from a pre-trained VGG16 network.
3. Design and implement an effective data augmentation pipeline for industrial images.
4. Evaluate the FAVAE model's performance on anomaly detection and localization tasks.
5. Analyze the model's robustness across different types of anomalies in MVTec AD datasets.

Chapter 5

Methodology

5.1 Dataset

The MVTec Anomaly Detection dataset [1] is used in this study. It comprises 5 categories of industrial objects and textures, including bottle, grid, pill, screw and toothbrush. Each category contains normal (defect-free) images for training and both normal and defective images for testing.

We utilize a custom MVTec AD Dataset class to load and manage the dataset. This class handles the organization of images into training and testing sets, as well as the association of ground truth masks for defective samples.

5.2 Data Preprocessing and Augmentation

Data preprocessing and augmentation play crucial roles in enhancing the model's ability to learn robust representations and generalize well to unseen data. Our approach includes several key steps:

5.2.1 Image Resizing and Normalization

All images are resized to a uniform size of 128x128 pixels. This standardization ensures consistent input dimensions for the model and reduces computational overhead. The images are then converted to tensors and normalized to the range [0, 1].

5.2.2 Data Augmentation

To increase the diversity of our training data and improve model robustness, we apply the following augmentation techniques:

- **Random Rotation**: Images are randomly rotated within a range of ± 15 degrees. This helps the model become invariant to slight orientation changes.
- **Random Cropping**: After rotation, images may be randomly cropped to maintain the target size of 128x128 pixels. This introduces variation in object positioning within the image.
- **Horizontal and Vertical Flipping**: Images have a 30% chance of being flipped horizontally and a 30% chance of being flipped vertically. This augmentation helps the model learn features that are invariant to these transformations.

The augmentation process is controlled by several parameters, including the probability of applying each transformation and the intensity of the rotation. These parameters can be fine-tuned to optimize the augmentation strategy for each specific object category.

5.2.3 Background Masking

To help the model focus on relevant features and reduce the impact of background variations, we implement a background masking technique. This involves:

- Converting images to grayscale (if not already).
- Applying thresholding to separate the object from the background.
- Filling holes in the resulting mask to create a more coherent object region.

This preprocessing step can significantly improve the model's ability to detect anomalies on the object surface by reducing the influence of background noise.

By combining these preprocessing and augmentation techniques, we create a rich and diverse dataset that helps our model learn robust representations of normal patterns in industrial objects. This forms a solid foundation for effective anomaly detection and localization in subsequent stages of our methodology.

5.3 Model Architecture

The core of our approach is a Variational Autoencoder (VAE) specially designed for processing MVTec AD datasets. The VAE consists of an encoder and a decoder, both implemented using convolutional neural networks (CNNs). Here are the key components of our model:

5.3.1 Encoder

The encoder network transforms the input image into a latent space representation. It consists of:

- Seven convolutional layers with increasing channel depths ($128 \rightarrow 128 \rightarrow 256 \rightarrow 256 \rightarrow 512 \rightarrow 512 \rightarrow 32$)
- Kernel sizes alternating between 4x4 (with stride 2 for downsampling) and 3x3
- Batch normalization after each convolutional layer
- LeakyReLU activation functions with a negative slope of 0.2
- A final convolutional layer to produce the latent space parameters (mean and log-variance)

The encoder progressively reduces the spatial dimensions from 128x128 to 1x1, while increasing the feature depth.

5.3.2 Latent Space

The latent space(z) has a dimension of 100. We use the reparameterization trick to enable backpropagation through the sampling process:

- The encoder outputs two vectors: mean (μ) and log-variance ($\log \sigma^2$)
- During training, we sample from this distribution: $z = \mu + \epsilon \cdot \exp(0.5 \cdot \log \sigma^2)$, where $\epsilon \sim \mathcal{N}(0, I)$
- During inference, we simply use the mean: $z = \mu$

5.3.3 Decoder

The decoder network reconstructs the input image from the latent representation. It mirrors the encoder structure:

- Seven transposed convolutional layers with decreasing channel depths (512 $\rightarrow$ 512 $\rightarrow$ 512 $\rightarrow$ 256 $\rightarrow$ 256 $\rightarrow$ 128 $\rightarrow$ reconstructed image)
- Kernel sizes and strides matching the encoder for appropriate upsampling
- Batch normalization and LeakyReLU activations after each layer, except the final output
- A sigmoid activation function at the output to produce pixel values in the range [0, 1]

5.3.4 Adapter Modules

We include adapter modules that can potentially augmented features from different layers of the network. These consist of two 1x1 convolutional layers with an intermediate ReLU activation.

5.4 Training Procedure

The training procedure for our FAVAE model involves the following key components:

5.4.1 Loss Function

We use a combination of three loss terms:

- Reconstruction Loss: Mean Squared Error (MSE) between the input and reconstructed images

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

- KL Divergence Loss: To enforce a standard normal distribution in the latent space

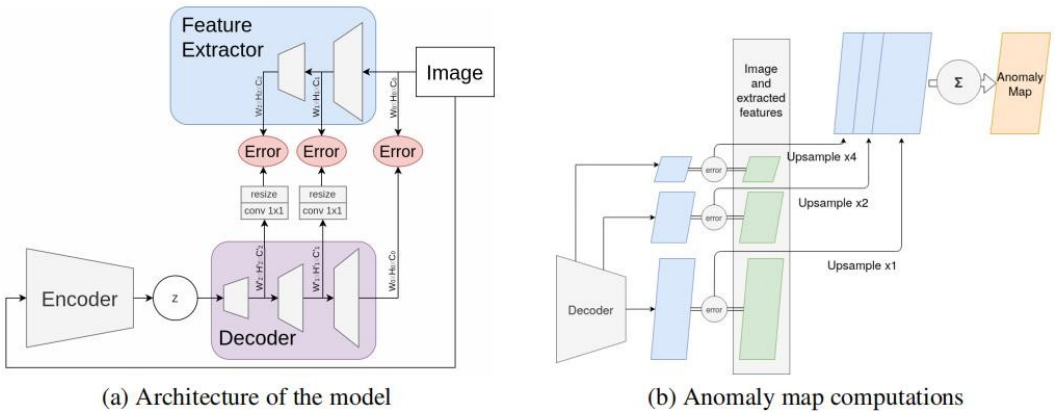
$$D_{\text{KL}} \left(\mathcal{N}(\mu, \sigma^2) \parallel \mathcal{N}(0, I) \right) = -\frac{1}{2} \sum_{j=1}^{D} \left(1 + \log \sigma_j^2 - \mu_j^2 - \sigma_j^2 \right)$$

- Feature Augmented Loss: MSE between the adapted features from the VAE decoder and the corresponding features from a pre-trained VGG16 or VGG19 network

The total loss is computed as:

$$\mathcal{L} = \mathcal{L}\text{MSE} + \alpha \cdot \mathcal{L}\text{KLd} + \mathcal{L}\text{Feature}$$

where α is a weighting factor to balance the KL divergence term.



(a) Architecture of the model (b) Anomaly map computations

Figure 5.1: An Illustration of the FAVAE model structure

5.4.2 Optimization

We use the Adam optimizer with the following parameters:

- Learning rate: 0.001
- Weight decay: 1e-5

5.4.3 Training Loop

For each epoch:

1. Forward pass: Input images are passed through the encoder and decoder
2. Extract features from intermediate layers of the decoder
3. Extract features from corresponding layers of the pre-trained VGG16 network
4. Compute the reconstruction loss, KL divergence, and feature augmented loss
5. Backpropagate the total loss
6. Update model parameters using the optimizer

5.4.4 Validation

After each training epoch, we evaluate the model on a validation set to monitor its performance and prevent overfitting. We use an early stopping mechanism with a patience of 20 epochs to halt training if the validation loss doesn't improve.

5.4.5 Feature Extraction and Augmented

We use a pre-trained VGG16 and VGG19 network as a teacher model to guide the feature learning process:

- Features are extracted from specific layers of the VGG16 and VGG19 network (layers 7, 14, and 21)
- Corresponding features are extracted from the VAE decoder (layers 10, 16, and 22)

- Adapter modules augment the dimensionality of the VAE features with the VGG features
- The MSE between augmented features contributes to the total loss

5.4.6 Data Augmentation

During training, we apply various data augmentation techniques to increase the diversity of our training data:

- Random rotation ($\pm$15 degrees)
- Random cropping
- Horizontal and vertical flipping (30)

5.4.7 Hyperparameters

Key hyperparameters in our training procedure include:

- Batch size: 32
- Number of epochs: 100 (maximum, subject to early stopping)
- KL divergence weight (α): 1.0
- Learning rate: 0.001
- Weight decay: 1e-5
- Validation ratio: 0.2 (20%of data used for validation)
- Image resize and crop size: 128x128 pixels

5.4.8 Monitoring and Visualization

Every 10 epochs, we save sample reconstructions of both validation and test images to visually inspect the model's progress. We also maintain a detailed log of training and validation losses for post-training analysis. This training procedure leverages the strengths of VAEs while incorporating guidance from a pre-trained network, aiming to learn more robust and discriminative features for anomaly detection in industrial images.

Chapter 6

Experiments and Results

6.1 Evaluation Protocol

To evaluate the performance of our FAVAE model, we utilize pixel-wise AUROC. The experimental setup details are as follows:

6.1.1 Hardware and Software

- Hardware: Training and evaluation were conducted on google colab.
- Software: The model was implemented in PyTorch. Data preprocessing and augmentation were done using the torchvision library. The pre-trained VGG16 and VGG19 network was sourced from the torchvision.models module.

6.2 Results

Our experiments demonstrate that the FAVAE model, when guided by features from a pre-trained VGG16 and VGG19 network, achieves superior anomaly detection and localization performance compared to baseline VAE models. Below is a detailed analysis of the results across different object categories:

Datasets	Pixel AUROC		
	VAE	VAE(VGG16)	VAE(VGG19)
Grid	0.65	0.901	0.883
Toothbrush	0.67	0.981	0.941
Pill	0.32	0.93	0.893
Screw	0.56	0.986	0.975
Bottle	0.45	0.957	0.931

6.2.1 Analysis

The results in Table 6.1 highlight the effectiveness of incorporating feature augmented with pre-trained VGG networks into the FAVAE model. Specifically:

- Overall Improvement: FAVAE (VGG16) consistently outperforms both the VAE-only model and the VAE (VGG19) across all object categories. This indicates that the feature guidance from VGG16 is more beneficial compared to VGG19 and the baseline model.
- Object-Specific Performance: All the objects shows ROCAUC improvements substantial compared to the baseline.

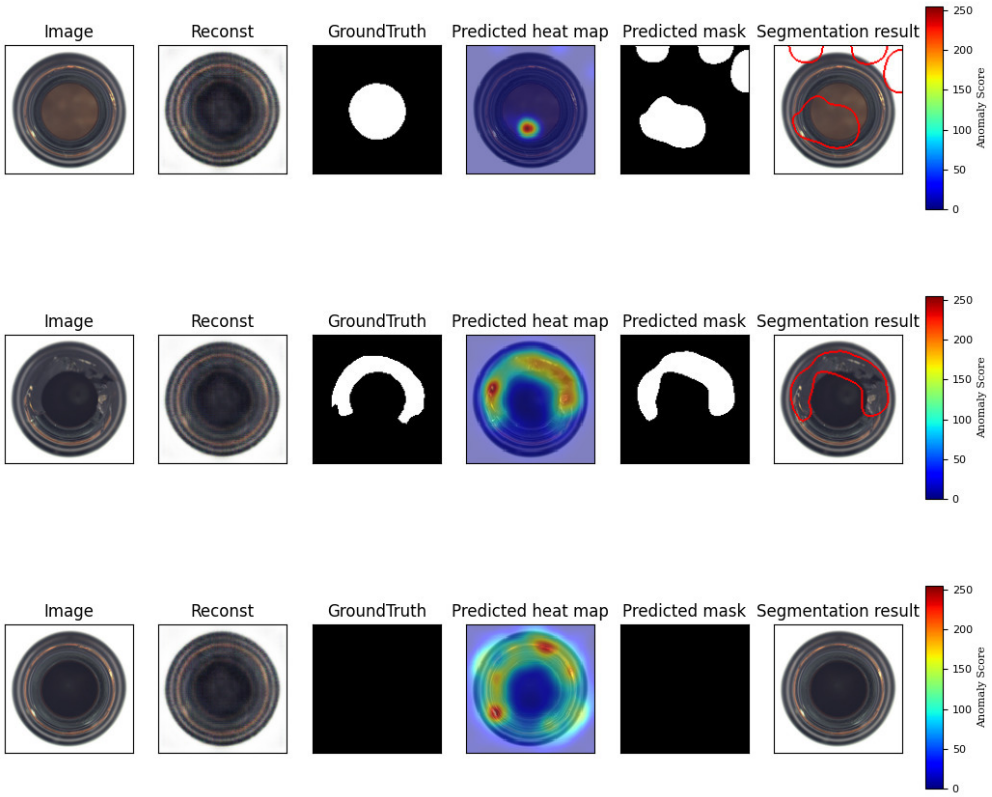


Figure 6.1: Bottle anomaly detection and localization result

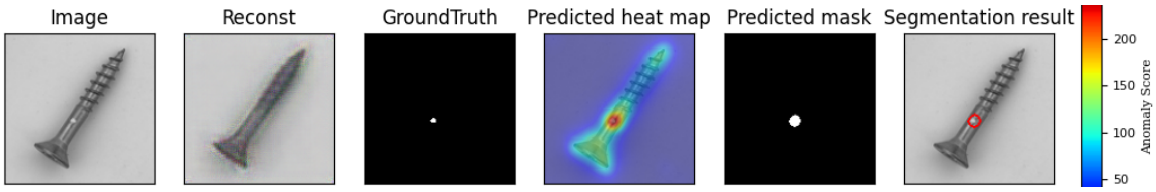
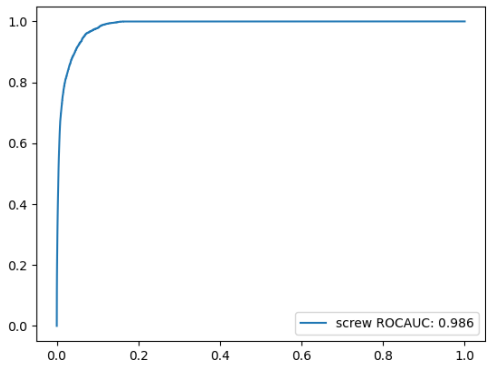
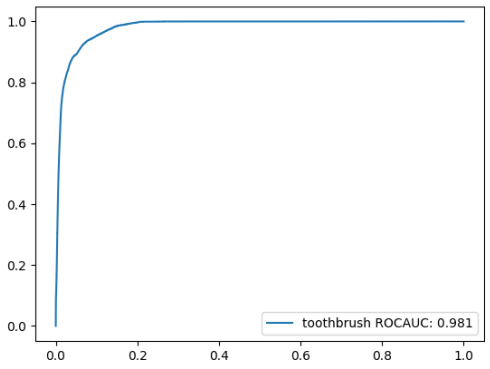


Figure 6.2: Screw anomaly detection and localization result

- Consistency: The performance gains are consistent across different categories, suggesting that the feature augmented technique generalizes well to various types of industrial images.
- Model Comparison: Comparing VGG16 and VGG19-guided models, VGG16 consistently shows slightly better performance. This could be attributed to differences in the architectures and the specific features learned by each network.



(a) Screw ROCAUC



(b) Toothbrush ROCAUC

Figure 6.3: An illustratin for ROAUC measurement

6.2.2 Visual Inspection

Every 10 epochs, sample reconstructions of validation and test images were saved to visually inspect the model's progress. These visual checks helped in ensuring that the model was learning meaningful features and not overfitting to the training data.

The experimental results confirm that our FAVAE model, leveraging feature augmented with a pre-trained VGG16 network, achieves improved anomaly detection and localization performance. This approach proves effective across various industrial objects, providing a robust solution for anomaly detection tasks.

Chapter 7

Discussions and Future Work

7.1 Discussion

The results of our experiments demonstrate that incorporating feature augmented from a pre-trained VGG16 and VGG19 network into the FAVAE model significantly enhances its ability to detect and localize anomalies in industrial images. The following points summarize key insights and implications:

7.1.1 Feature Augmented Effectiveness

Augmenting features from the VAE decoder with those from the VGG16 and VGG19 network helps the model learn more discriminative features, improving its performance in detecting subtle anomalies. This indicates the importance of leveraging pre-trained networks that have learned rich, hierarchical feature representations.

7.1.2 Comparison of VGG16 and VGG19

While both VGG16 and VGG19-guided models outperform the baseline VAE model, VGG16 consistently shows better performance. This could be due to the specific feature extraction capabilities of VGG16, which might be better suited for the anomaly detection task in industrial images.

7.1.3 Performance Across Object Categories

The consistent performance improvement across all object categories suggests that the proposed method generalizes well.

7.1.4 Training Efficiency and Stability

Using the Adam optimizer with a carefully selected learning rate and weight decay ensures stable and efficient training. The early stopping mechanism with a patience of 20 epochs helps in preventing overfitting, allowing the model to generalize better to unseen data.

7.2 Future Work

Despite the promising results, there are several areas for further investigation and improvement:

7.2.1 Exploring Other Pre-trained Networks

Future research could explore the use of other pre-trained networks such as ResNet or EfficientNet for feature augmented. These networks have different architectures and might provide complementary feature representations that could further improve performance.

7.2.2 Enhancing Feature Augmented Mechanism

Improving the feature augmented mechanism, perhaps through the use of more sophisticated augmented techniques or loss functions, could yield even better results. Techniques such as attention mechanisms or adversarial training could be explored.

7.2.3 Extended Data Augmentation Techniques

Incorporating additional data augmentation techniques, such as color jittering, Gaussian noise addition, and elastic transformations, could increase the robustness of the model to various types of image distortions and variations.

7.2.4 Real-time Anomaly Detection

Adapting the model for real-time anomaly detection in industrial settings is another area for future work. This would involve optimizing the model inference speed and integrating it with real-time monitoring systems.

7.2.5 Transfer Learning for Other Domains

Applying the proposed FAVAE model to other domains beyond industrial images, such as medical imaging or autonomous driving, could be explored. This would test the model's adaptability and robustness across different types of data.

Chapter 8

Conclusion

In this project, we introduced the Feature Augmented Variational Autoencoder (FAVAE) for anomaly detection in industrial images. Our approach leverages the power of Variational Autoencoders (VAEs) augmented with high-level feature representations from a pre-trained VGG16 and VGG19 networks, leading to significant improvements in both anomaly detection and localization performance over baseline models.

Experimental results demonstrate that the FAVAE effectively learns discriminative features, yielding consistent performance gains across various object categories. This indicates the robustness and generalizability of our model, making it suitable for a wide range of industrial applications.

Looking forward, future research will explore integrating other pre-trained networks to further enhance the feature augmentation process and adapt the model for real-time anomaly detection applications.

Bibliography

[1] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. MVTec AD – A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In CVPR, 2019. https://www.mvtec.com/company/research/datasets/mvtec-ad/

[2] Lukas Ruff et al. Deep one-class classification. In International conference on machine learning, 2018.

[3] Bo Zong et al. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In International Conference on Learning Representations, 2018.

[4] Thomas Schlegl et al. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In International conference on information processing in medical imaging, 2017.

[5] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.

[6] AutoEncoder-SSIM-for-unsupervised-anomaly-detectionhttps://github.com/plutoyuxie/AutoEncoder-SSIM-for-unsupervised-anomaly-detection-

[7] Anomaly localization by modeling perceptual features https://arxiv.org/pdf/2008.05369.