Joint use of skip connections and synthetic corruption for anomaly detection with autoencoders

Anne-Sophie Collin and Christophe De Vleeschouwer

Abstract - In industrial vision, the anomaly detection problem can be addressed with an autoencoder trained to map an arbitrary image, i.e. with or without any defect, to a clean image, i.e. without any defect. In this approach, anomaly detection relies conventionally on the reconstruction residual or, alternatively, on the reconstruction uncertainty. To improve the sharpness of the reconstruction, we consider an autoencoder architecture with skip connections. In the common scenario where only clean images are available for training, we propose to corrupt them with a synthetic noise model to prevent the convergence of the network towards the identity mapping, and introduce an original Stain noise model for that purpose. We show that this model favors the reconstruction of clean images from arbitrary real-world images, regardless of the actual defects appearance. In addition to demonstrating the relevance of our approach, our validation provides the first consistent assessment of reconstruction-based methods, by comparing their performance over the MVTec AD dataset [1], both for pixel- and image-wise anomaly detection. Our implementation is available at `https://github.com/anncollin/AnomalyDetection-Keras`.

1 Introduction

Anomaly detection can be defined as the task of identifying all diverging samples that does not belong to the distribution of regular, also named clean, data. This task could be formulated as a supervised learning problem. Such an approach uses both clean and defective examples to learn how to distinguish these two classes or even to re-

Anne-Sophie Collin
UCLouvain, Belgium, e-mail: `anne-sophie.collin@uclouvain.be`

Christophe De Vleeschouwer
UCLouvain, Belgium, e-mail: `christophe.devleeschouwer@uclouvain.be`

fine the classification of defective samples into a variety of subclasses. However, the scarcity and variability of the defective samples make the data collection challenging and frequently produce unbalanced datasets [2]. To circumvent the above-mentioned issues, anomaly detection is often formulated as an unsupervised learning task. This formulation makes it possible to either solve the detection problem itself or to ease the data collection process required by a supervised approach.
Anomaly detection has a broad scope of application, which implies that processed data can differ in nature. It typically corresponds to speech sequences, time series, images or video sequences [3]. Here, we consider the automated monitoring of production lines through visual inspection to detect defective samples. More specifically, we are interested in identifying abnormal structures in a manufactured object based solely on the analysis of one image of the considered item. Computer vision sensors offer the opportunity to be easily integrated in a production line without disturbing the production scenarios [4]. To handle such high-dimensional data, Convolutional Neural Networks (CCNs) provide a solution of choice, due to their capacity to extract rich and versatile representations.

The unsupervised anomaly detection framework considered in this work is depicted in Figure 1. It builds on the training of an autoencoder to project an arbitrary image onto the clean distribution of images (blue block). The training set is constituted exclusively of clean images. Then, defective structures can be inferred from the reconstruction (red block), following a traditional approach based on the residual [2], or even from an estimation of the prediction uncertainty [5].
During training, the autoencoder is constrained to minimize the reconstruction error of clean structures in the images. Several loss functions, presented later in Section 2, can be considered to quantify this reconstruction error. With the objective of building our method on the Mean Squared Error (MSE) loss for its simplicity and widespread usage, we propose a new non-parametric approach that addresses the standard issues related to the use this loss function. To enhance the sharpness of the reconstruction, we consider an autoencoder equipped with skip connections, which allow the information to bypass the bottleneck. In order to prevent systematic transmission of the image structures through these links, the network is trained to reconstruct a clean image out of a corrupted version of the input, instead of an unmodified version of it. As discussed later, the methodology used to corrupt the training images has a huge impact on the overall performances. We introduce a new synthetic model, named Stain, that adds an irregular elliptic structure of variable color and size to the input image. Despite its simplicity, the Stain model is by far the best performing compared to the scene-specific corruption investigated in a previous study [6]. Our Stain model has the double advantage of performing consistently better, while being independent of the image content. We demonstrate that adding skip connections to the autoencoder architecture when simultaneously corrupting the training clean images with our Stain noise model addresses the blurry reconstruction issue related to the use of the MSE loss.

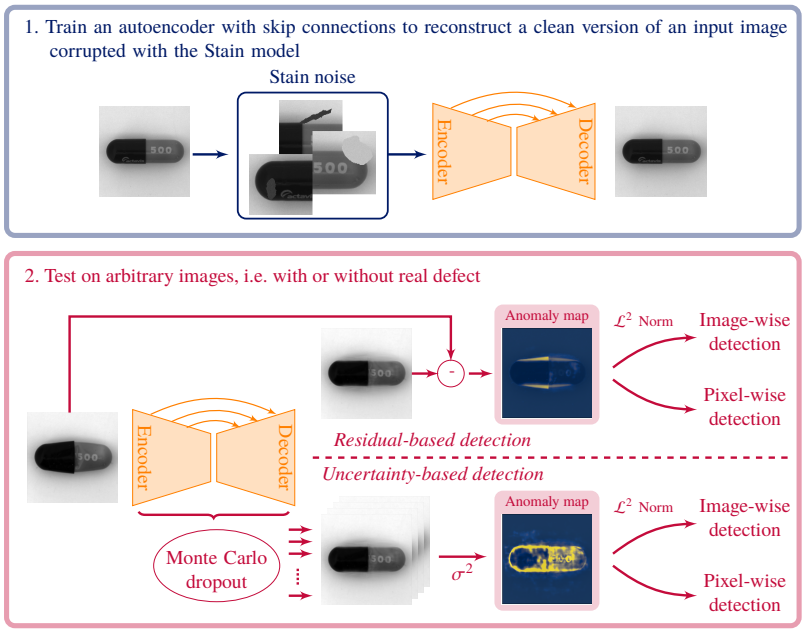


Fig. 1 We improve the quality of the reconstructed images by training an autoencoder with skip connections on corrupted images. *1. Blue block.* Corrupting the training images with our Stain noise model avoids the convergence of the network towards an unwanted identity operator. *2. Red block.* The two anomaly detection strategies. In the upper part, the anomaly map is generated by subtracting the input image from its reconstruction. In the lower part, the anomaly map is estimated by the variance between 30 reconstructions inferred with Monte Carlo dropout (MCDropout) [7]. It relies on the hypothesis that structures that are not seen during training (defective areas) correlate with higher reconstruction uncertainty.

The present work extends our conference paper [8] by providing an extensive study of the internal statistics of the network. This analysis reveals the natural trend of the network to distinguish between the representations of regular and irregular samples, which is even more explicit in the bottleneck layer. Moreover, we show that, when applied simultaneously, the two modifications of the autoencoder-based reconstruction framework studied in this work, namely :

1. the corruption of the training images with our Stain-noise model, and
2. the addition of skip connections to the autoencoder architecture,

lead to a better separation of the internal representations associated to initial and reconstructed versions of irregular samples than regular ones. This phenomenon is observed even though the network has not been explicitly trained to separate these two classes.

In Section 2, we provide an overview of previous reconstruction-based methods addressing the anomaly detection problem, and motivate the use of the MSE loss to

train our auto-encoder. Details of our method, including network architecture and the Stain noise model description, are provided in Section 3. In Section 4, we provide a comparative study of residual- and uncertainty-based anomaly detection strategies, both at the image and pixel level. This extensive comparative study demonstrates the benefit of our proposed framework, combining skip connections and our original corruption model. Section 5 further investigates and compares the internal representations of clean and defective images in an autoencoder with and without skip connections. This analysis provides insight into the variety of performance observed in our comparative study, and allows to develop intuition regarding the obtained results. Moreover, this study highlights the discrepancies observed across image categories of the MVTec AD dataset, and raises questions for future research. Section 6 concludes.

2 Related Work

Anomaly detection is a long-standing problem that has been considered in a variety of fields [2, 3] and the reconstruction-based approach is one popular way to address the issue. In comparison to other methods for which the detection of abnormal samples is performed in another domain than the image [9–13], reconstruction-based approaches offer the opportunity to identify the pixels that lead to the rejection of the image from the normal class. This section presents a literature review organized into three subsections, each one focusing on the main issues encountered when working with a reconstruction-based approach.

Low contrast defects detection. Conventional reconstruction-based methods infer anomaly based on the reconstruction error between an arbitrary input and its reconstructed version. It assumes that clean structures are perfectly conserved while defective ones are replaced by clean content. However, when a defect contrasts poorly with its surroundings, replacing abnormal structures with clean content does not lead to a sufficiently high reconstruction error. In such cases, this methodology reaches the limit of its underlying assumptions. A previous study [5] detected anomalies by quantifying the prediction uncertainty with MCDropout [7] instead of the reconstruction residual.

Reconstruct sharp structures. To obtain a clean reconstruction out of an arbitrary image, an autoencoder is trained on clean images to perform an image-to-image identity mapping under the minization of a loss function. The bottleneck forces the network to learn a compressed representation of the training data that is expected to regularize the reconstruction towards the normal class. In the literature, the use of the MSE loss to train an hourglass CNN, without skip connections, has been criticized for its trend to produce blurry output images [14, 15]. Since anomaly detection is based on the reconstruction residual, this behavior is detrimental because it alters the clean structures of an image as well as the defective ones.

A lot of effort has been made to improve the quality of the reconstructed images by the introduction of new loss functions. In this spirit, unsupervised methods based on Generative Adversarial Networks (GANs) have emerged [16–20]. If GANs are known for their ability to produce realistic high-quality synthetic images [21], they have major drawbacks. Usually, GANs are difficult to train due to their trend to converge towards mode collapse [22]. Moreover, in the context of anomaly detection, some GAN-based solutions fail to exclude defective samples from the generative distribution [19] and require an extra optimization step in the latent space during inference [17]. This process ensures that the defective structures of the input image are replaced by clean content. Performances of AnoGAN [17] over the MVTec AD dataset have been reported by Bergmann et al. [1]. Those are significantly lower than the method proposed in this work.
To improve the sharpness of the reconstruction, Bergmann et al. proposed a loss derived from the Structural SIMilarity (SSIM) index [15]. The use of the SSIM loss has been motivated by its ability to produce well looking images from a human perceptual perspective [14, 23]. The SSIM have shown some improvement over the MSE loss for the training of an autoencoder in the context of anomaly detection. However, the SSIM loss formulation does not generalize to color images and is parametric. Traditionally, these hyper-parameters are tuned based on a validation set. However, in a real-life scenarios of anomaly detection, samples with real defects are usually not available. For this reason, our paper focuses on the MSE rather than on the parametric SSIM.

Prevent the reconstruction of defective structures. It is usually expected that the compression induced by the bottleneck is sufficient to regularize the reconstruction towards the clean distribution of images. In practice, the autoencoder is not explicitly constrained to not reproduce abnormal content and often reconstructs defective structures. A recent method proposed to mitigate this issue by iteratively projecting the arbitrary input towards the clean distribution of images. The projection is constrained to be similar, in the sense of the $\mathcal{L}^1$ norm, to the initial input [24]. Instead of performing this optimization in the latent space as made with AnoGAN [17], they propose to find an optimal clean input image. If this practice enhances the sharpness of the reconstruction, the optimization step is resource consuming.
Also, the reconstruction task can be formulated as an image completion problem [25, 26]. To make the inference and training phases consistent, it is assumed that the defects are entirely contained in the mask during inference, which limits the practical usage of the method. Random inpainting maks have been considered to deal with this issue [27]. Mei et al. [28] also proposed to use a denoising autoencoder to reconstruct training images corrupted with salt-and-pepper noise. However they did not discuss the gain brought by this modification, and only considered it for an hourglass CNN, without skip connections.
The reconstruction of clean structures has also been promoted by constraining the latent representation. Practically, Gong et al. [29] learned a dictionary to describe the latent representation of clean samples based on a sparse linear combination of dictionary elements. In contrast, defective samples are assumed to require more

complex combinations of the dictionary items. Based on this hypothesis, enforcing sparsity during inference prevents the reconstruction of defective structures. In a similar spirit, Wang et al. [30] considered the use of a VQ-VAE [31]. Their method incorporates a specific autogressive model in the latent space which can be used to enforce similarity between encoded vectors of the training and the test sets, thereby preventing the reconstruction of defective structures.

The methodology proposed in this work presents a simple approach to enhance the sharpness of the reconstructed images. The skip connections allow the preservation of high frequency information by bypassing the bottleneck. However, we show that this practice penalizes anomaly detection when the model is trained to perform identity mapping on uncorrupted clean images. Nevertheless, the introduction of an original noise model allows to significantly improve the anomaly detection accuracy for the skipped architecture, which eventually outperforms the conventional one in many real-life cases. Also, we compare anomaly detection based on the reconstruction residual or uncertainty estimation. This second option appears to be of particular interest for the detection of low contrast defective structures.

3 Our anomaly detection framework

Our method addressed anomaly detection based on the regularized reconstruction performed by an autoencoder. This section presents the different components of our approach, ranging from the training of the autoencoder to the strategies considered to detect defects based on the reconstruction residual or the reconstruction uncertainty.

3.1 Model configuration

The reconstruction of a clean version of any input image is based on a CNN. Our architecture, referred to as **Autoencoder with Skip connections (AESc)** and shown in Figure 2, is a variant of U-Net [32]. AESc takes input images of size 256×256 and projects them onto a latent space of $4 \times 4 \times 512$ dimension. The projection towards the lower dimensional space is performed by six consecutive convolutional layers strided by a factor two. The back projection is performed by six layers of convolution followed by an upsampling operator of factor two. All convolutions have a 5×5 kernel. Unlike the original U-Net version, our skip connections perform an addition, not a concatenation, of feature maps of the encoder to the decoder.
For the sake of comparison, we also consider the **Autoencoder (AE)** network which follows the same architecture but from which we removed the skip connections.

All models have been trained during 250 epochs to minimize the MSE loss over batches constituted of 16 images. We used the Adam optimizer [33] with an initial

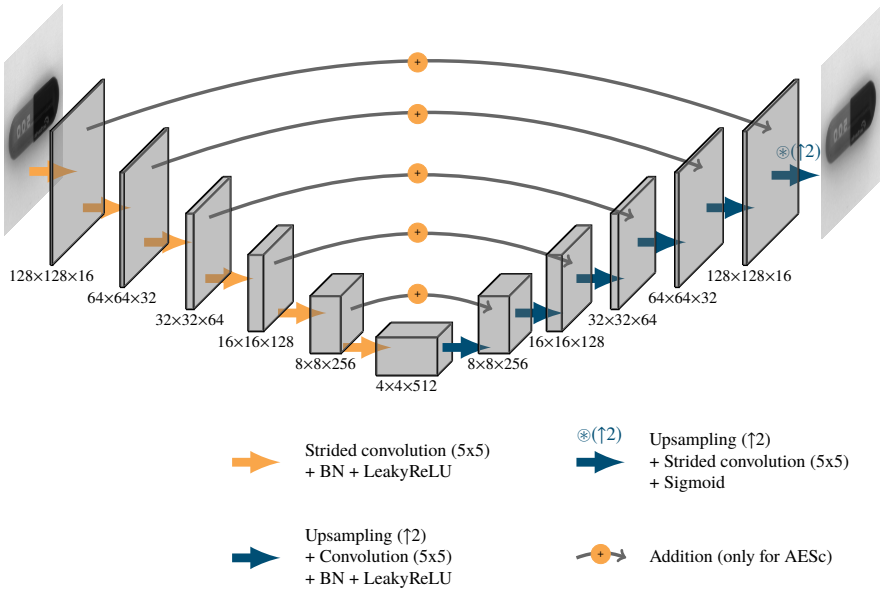


Fig. 2 AESc architecture performing the projection of an arbitrary 256×256 image towards the distribution of clean images with the same dimension. Note that the AE architecture shares the same specifications with the exception of the skip connections that have been removed.

learning rate of 0.01. This learning rate is is decreased by a factor of 2 when the PSNR over a validation set, constituted by 20% of the training images, reaches a plateau for at least 30 epochs. No additional data augmentation than the synthetic corruption model presented in Section 3.2 is applied on the training set.

3.2 Corruption model

Ideally, the autoencoder should preserve clean structures while modifying those that are not. To this end, it is wanted that defective structures are excluded from the generative model, eventhough the training task is an identity mapping. Due to the impossibility of collecting pairs of clean and defective versions of the same sample, we propose to introduce synthetic corruption during training to explicitly constrain the autoencoder to remove this additive noise. Our **Stain** noise model, illustrated in Figure 1 and explained in Figure 3, corrupts images by adding a structure whose color is randomly selected in the grayscale range and whose shape is an ellipse with irregular edges.

The intuition behind the definition of this noise model is that occluding large area of the training images is a data augmentation procedure that helps to improve the

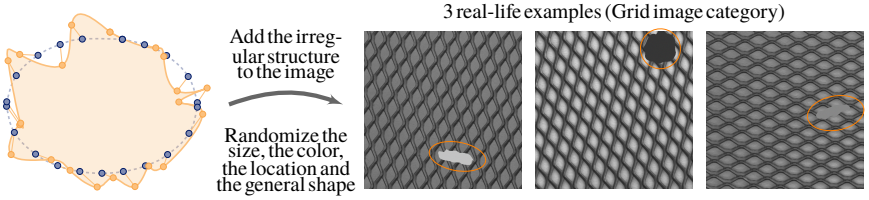


Fig. 3 The Stain noise model is a cubic interpolation between 20 points (orange dots), arranged in ascending order of polar coordinates, located around the border of an ellipse of variable size (blue line). The axes of the ellipse are comprised between 1 and 12% of the smallest image dimension and its eccentricity is randomly initialized.

network training [34, 35]. Due to the skip connections in our network architecture, this form of data augmentation is essential to avoid the convergence of the model towards the identity operator. However, we noticed that the use of regular shapes, like a conventional ellipse, leads to overfitting to this additive noise structure, as also pointed out in a context of inpainting with rectangular holes [36].

3.3 Anomaly detection strategies

We compare two approaches to obtain the anomaly map representing the likelihood that a pixel is abnormal. On the one hand, the **residual-based** approach evaluates the abnormality by measuring the absolute difference between the input image $\mathbf{x}$ and its reconstruction $\hat{\mathbf{x}}$. On the other hand, the **uncertainty-based** approach relies on the intuition that structures that are not seen during training, i.e. the anomalies, will correlate with higher uncertainties. This is estimated by the variance between 30 output images inferred with the MCDropout technique. Our experiments revealed that more accurate detection is obtained by applying an increasing level of dropout for deepest layers. More specifically, the dropout levels are [0, 0, 10, 20, 30, 40] percent for layers ranging from the highest spatial resolution to the lowest.

Out of the anomaly map, it is either possible to classify the entire image as clean/defective or to classify each pixel as belonging to a clean/defective structure. In the first case, referred to as **image-wise detection**, it is common to compute the $\mathcal{L}^p$ norm of the anomaly map given by

$$\mathcal{L}^p(\mathbf{x}, \hat{\mathbf{x}}) = \left(\sum_{i=0}^{m} \sum_{j=0}^{n} |\mathbf{x}_{i,j} - \hat{\mathbf{x}}_{i,j}|^p \right)^{1/p} \tag{1}$$

with $\mathbf{x}_{i,j}$ denoting the pixel belonging to the i^{th} row and the j^{th} column of the image $\mathbf{x}$ of size $m \times n$. Based on our experiments, we present results obtained for $p = 2$

since they achieve the most stable accuracy values across the experiments. Hence, all images for which the $\mathcal{L}^2$ norm of the abnormality map exceeds a chosen threshold are considered as defective. In the second case, referred to as **pixel-wise detection**, the threshold is applied directly on each pixel value of the anomaly map.

To perform image-wise or pixel-wise anomaly detection, a threshold has to be determined. Since this threshold value is highly dependent on the application, we present the performances in terms of Area Under the receiver operating characteristic Curve (AUC), obtained by varying over the full range of threshold values.

4 Anomaly detection on the MVTec AD dataset

Experiments have been conducted on grayscale images of the MVTec AD dataset [1], containing five categories of textures and ten categories of objects. In this dataset, defects are real and have various appearance. Their location is defined with a binary segmentation mask. All images have been scaled to a 256×256 size. Anomaly detection is performed at this resolution.

4.1 AESc + Stain: qualitative and quantitative analysis

In this section, we compare qualitatively and quantitatively the results obtained with our AESc + Stain model for both image- and pixel-wise detection. This analysis focuses on the residual-based detection approach to emphasize the benefits of adding skip connections to the AE architecture. The comparison of the results obtained with residual- versus uncertainty-based strategies is discussed later in Section 4.2.

Qualitatively, Figure 4 reveals the trends of the AE and AESc models trained with and without the Stain noise corruption. On the one hand, the AE network produces blurry reconstructions as depicted by the overall higher residual intensities. If the global structure of the object (Cable and Toothbrush) are properly reconstructed, the AE network struggles to infer the finer details of the texture images (Carpet sample). On the other hand, the AESc model shows finer reconstruction of the image details depicted by a nearly zero residual over the clean areas of the images. However, when ASEc is trained without corruption, the model converges towards an identity operator, as revealed by the close-to-zero residuals of defective structures. The corruption of the training images with the Stain model alleviates this unwanted behavior by leading to high reconstruction residuals in defective areas while simultaneously keeping low reconstruction residuals in clean structures.

Quantitatively, Table 1 presents the image-wise detection performances obtained with the AESc and AE networks trained with and without our Stain noise model.

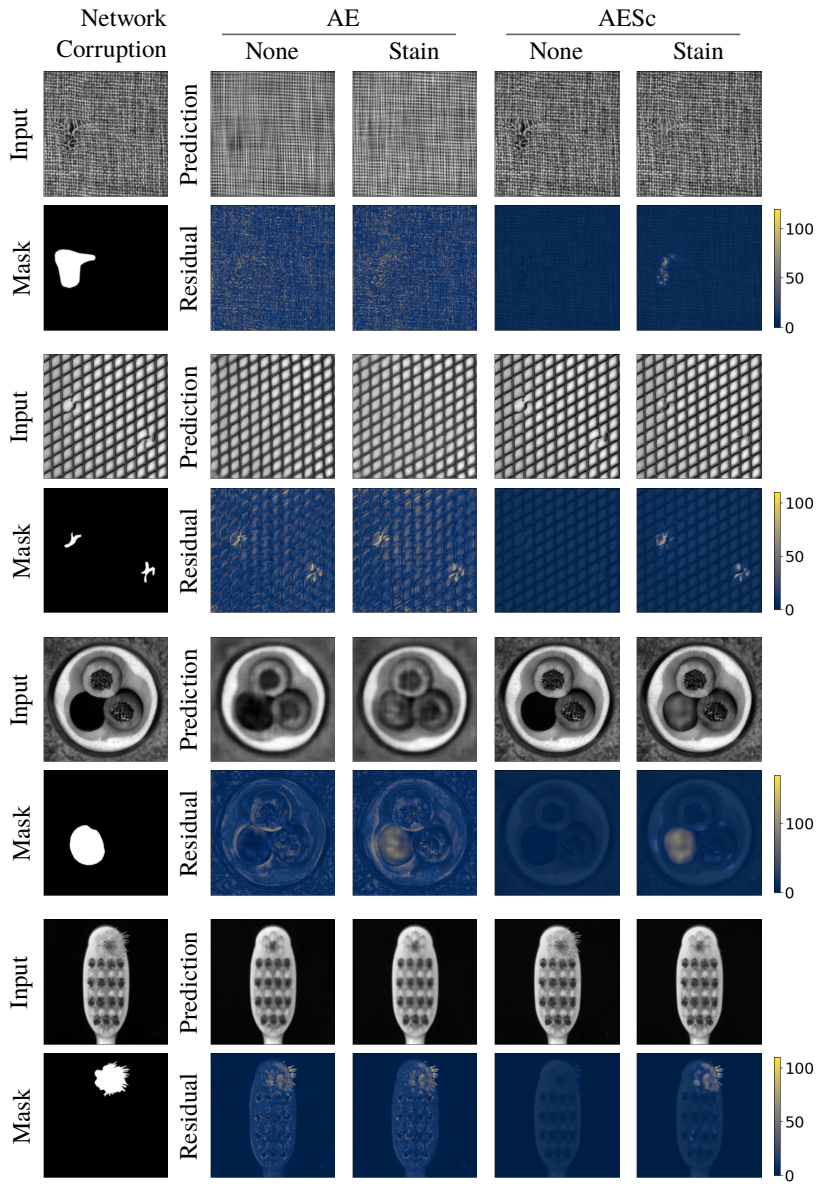


Fig. 4 Predictions obtained with the AESc and AE networks trained with and without our Stain noise model. Two defective textures are considered, namely a Carpet (first sample) and a Grid (second sample), as well as two defective objects, namely a Cable (third sample) and a Toothbrush (fourth sample). First column show the image fed in the networks and the mask locating the defect. Odd rows show the reconstructed images and even rows show the anomaly maps obtained with the residual-based strategy.

The last column provides a comparison with the ITEA method, introduced by Huang et al. [6]. ITAE is also a reconstruction-based approach which relies on an autoencoder with skip connections trained with images corrupted by random rotations and a graying operator (averaging of pixel value along the channel dimension) selected based on prior knowledge about the task.
This table highlights the superiority of our AESc + Stain noise model to solve the image-wise anomaly detection. The improvement brought by adding skip connections to an autoencoder trained with corrupted images is even more important for texture images than for object images. We also observe that, if the highest accuracy is consistently obtained with the residual-based approach, the uncertainty-based decision derived from the AESc + Stain model generally provides the second best (underlined in Table 1) performances among tested networks, attesting the quality of the AESc + Stain model for image-based decision.

Table 1 Image-wise detection AUC obtained with the residual- and uncertainty-based detection methods[a].

		Uncertainty				**Residual**				
	Network	**AE**		**AESc**		**AE**		**AESc**		**ITAE [6]**
	Corruption	None	Stain	None	Stain	None	Stain	None	Stain	
Textures	Carpet	0.41	0.30	0.44	0.80	0.43	0.43	0.48	**0.89**	0.71
	Grid	0.69	0.66	0.12	**0.97**	0.80	0.84	0.52	**0.97**	0.88
	Leather	0.86	0.57	0.88	0.72	0.45	0.54	0.56	**0.89**	0.87
	Tile	0.73	0.50	0.72	0.95	0.49	0.57	0.88	**0.99**	0.74
	Wood	0.87	0.86	0.78	0.78	0.92	0.94	0.92	**0.95**	0.92
	Mean[b]	0.71	0.58	0.59	0.84	0.62	0.66	0.67	**0.94**	0.82
Objets	Bottle	0.72	0.41	0.71	0.82	**0.98**	0.97	0.77	**0.98**	0.94
	Cable	0.64	0.48	0.52	0.87	0.70	0.77	0.55	**0.89**	0.83
	Capsule	0.55	0.49	0.44	0.71	**0.74**	0.64	0.60	**0.74**	0.68
	Hazelnut	0.83	0.60	0.68	0.90	0.90	0.88	0.85	**0.94**	0.86
	Metal Nut	0.38	0.33	0.41	0.62	0.57	0.59	0.24	**0.73**	0.67
	Pill	0.63	0.48	0.55	0.62	0.76	0.76	0.70	**0.84**	0.79
	Screw	0.45	0.77	0.13	0.80	0.68	0.60	0.30	0.74	**1.00**
	Toothbrush	0.36	0.44	0.51	0.99	0.93	0.96	0.78	**1.00**	**1.00**
	Transistor	0.67	0.59	0.55	0.90	0.84	0.85	0.46	**0.91**	0.84
	Zipper	0.44	0.41	0.70	0.93	0.90	0.88	0.72	**0.94**	0.80
	Mean[c]	0.57	0.50	0.52	0.82	0.80	0.79	0.60	**0.87**	0.84
	Global mean[d]	0.62	0.53	0.54	0.83	0.74	0.75	0.62	**0.89**	0.84

[a] For each row, the best performing approach is highlighted in boldface and the second best is underlined.
[b] Mean AUC obtained over the classes of images belonging to the texture categories.
[c] Mean AUC obtained over the classes of images belonging to the object categories.
[d] Mean AUC obtained over the entire dataset.

Table 2 presents the pixel-wise detection performances obtained with our approaches and compares them with the method reported in [1], referred to as AE_{L2}. This residual-based method relies on an autoencoder without skip connections, and provides state of the art performance in the pixel-wise detection scenario. Similarly to our AE model, AE_{L2} is trained to minimize the MSE of the reconstruction of images that are not corrupted with synthetic noise. AE_{L2} however differs from our AE model in several aspects, including a different network architecture, data augmentation, patch-based inference for the texture images, and anomaly map post-processing with mathematical morphology. Despite our efforts, in absence of public code, we have been unable to reproduce the results presented in [1]. Hence, our table just copy the results from [1]. For fair comparison between AE and AESc + Stain, the table also provides the results obtained with our AE, since our AE and AESc + Stain models adopt the same architecture (up to the skip connections) and the same training procedure.

Table 2 Pixel-wise detection AUC obtained with the residual- and uncertainty-based detection methods[a].

		Uncertainty				**Residual**				
	Network	**AE**		**AESc**		**AE**		**AESc**		**AE_{L2} [1]**
	Corruption	None	Stain	None	Stain	None	Stain	None	Stain	
Textures	Carpet	0.55	0.54	0.43	**0.91**	0.57	0.62	0.52	0.79	0.59
	Grid	0.52	0.49	0.50	**0.95**	0.81	0.82	0.57	0.89	0.90
	Leather	0.86	0.52	0.58	0.87	0.79	0.82	0.71	**0.95**	0.75
	Tile	0.54	0.50	0.53	**0.79**	0.45	0.54	0.62	0.74	0.51
	Wood	0.61	0.48	0.51	**0.84**	0.64	0.71	0.65	**0.84**	0.73
	Mean[b]	0.62	0.51	0.51	**0.87**	0.65	0.70	0.61	0.84	0.70
Objects	Bottle	0.68	0.63	0.64	**0.88**	0.85	**0.88**	0.47	0.84	0.86
	Cable	0.54	0.70	0.66	0.84	0.62	0.83	0.72	0.85	**0.86**
	Capsule	0.92	0.89	0.65	**0.93**	0.87	0.87	0.63	0.83	0.88
	Hazelnut	**0.95**	0.91	0.60	0.89	0.92	0.93	0.79	0.88	**0.95**
	Metal Nut	0.79	0.73	0.50	0.62	0.82	0.84	0.52	0.57	**0.86**
	Pill	0.82	0.82	0.61	**0.85**	0.81	0.81	0.64	0.74	**0.85**
	Screw	0.94	0.94	0.61	0.95	0.93	0.93	0.72	0.86	**0.96**
	Toothbrush	0.84	0.83	0.79	**0.93**	0.92	**0.93**	0.73	**0.93**	**0.93**
	Transistor	0.79	0.64	0.51	0.78	0.79	0.82	0.56	0.80	**0.86**
	Zipper	0.78	0.77	0.60	**0.90**	0.73	0.75	0.60	0.78	0.77
	Mean[c]	0.81	0.79	0.62	0.86	0.83	0.86	0.64	0.81	**0.88**
	Global mean[d]	0.74	0.69	0.58	**0.86**	0.77	0.81	0.63	0.82	0.82

[a] For each row, the best performing approach is highlighted in boldface and the second best is underlined.
[b] Mean AUC obtained over the classes of images belonging to the texture categories.
[c] Mean AUC obtained over the classes of images belonging to the object categories.
[d] Mean AUC obtained over the entire dataset.

In the residual-based detection strategy, our AESc + Stain method obtains similar performances as the AE_{L2} approach when averaged over all the image categories of the MVTec AD dataset. However, as already pointed in the image-wise detection scenario, AESc + Stain performs better with texture images and worse with object images. Regarding the decision strategy, we observe an opposite trend than the one encountered for image-wise detection: the uncertainty-based approach performs a bit better than the residual-based strategy when it comes to pixel-wise decisions. This difference is further investigated in the next section.

4.2 Residual- vs. uncertainty-based detection strategies

Figure 5 provides a visual comparison between residual- and uncertainty-based strategies. Globally, we observe that the reconstruction residual mostly correlates with the uncertainty. However, the uncertainty indicator is usually more widespread. This behavior can sometimes lead to a better coverage of the defective structures (Bottle and Pill) or to an increase of the number of false positive pixels that are detected (Carpet and Cable).
One important observation concerns the relationship between the detection of a defective structure and its contrast with its surroundings. In the residual-based approach, regions of an image are considered as defective if their reconstruction error exceeds a threshold. In the proposed formulation, the network is explicitly constrained to replace synthetic defective structures with clean content. No constraint is introduced regarding the contrast of the reconstructed structure and its surroundings. Hence, defects that are poorly contrasted lead to small residual intensities. On the contrary, the intensity of the uncertainty indicator does not depend on the contrast between a structure with the surroundings. For low contrast defects, it enhances their detection as illustrated (Bottle and Pill). On the contrary, it can deteriorate the location of high contrast defects for which the residual map is an appropriate anomaly indicator (Carpet and Cable). In theses cases, the sharp prediction obtained with the residual-based approach is preferred over the uncertainty-based one.

As reported in Section 4.1, we observe that the uncertainty-based detection perform generally worse than the residual-based approach for image-wise detection. We explain this drop of performance by an increase of the intensities of the uncertainty maps inferred from the clean images belonging to the test set. As the image-wise detection is based on the $\mathcal{L}^2$ norm of the anomaly map, the lowest the anomaly maps of clean images, the better the detection of defective images. For image-wise detection, the performances are less sensitive to the optimal coverage of the defective area as long as the overall intensity of the clean anomaly maps is low.
On the contrary, the uncertainty-based strategy improves the pixel-wise detection of the AESc + Stain model. For this use case, a better coverage of the defective structure is crucial. As previously mentioned, AESc + Stain model used usually leads to

reconstruction residual constituted of sporadic spots and misses low contrast defects. The uncertainty-based strategy compensates these two issues.

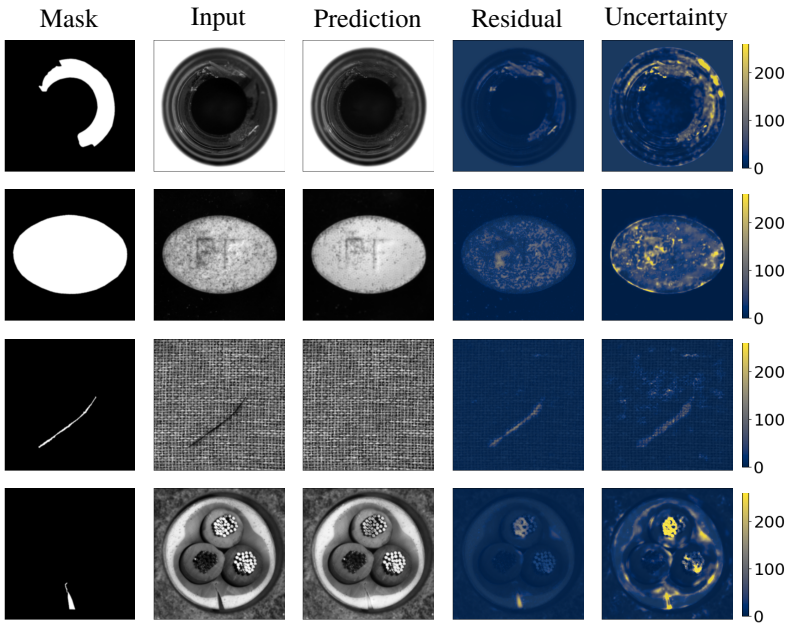


Fig. 5 Predictions obtained with the AESc network trained with our Stain noise model. One defective texture is considered, namely a Carpet (third row) as well as three defective objects, namely a Bottle (first row), a Pill (second row) and a Cable (fourth row). From left to right, columns represent the ground-truth, the image fed to the network, the prediction (without MCDropout), the reconstruction residual and the reconstruction uncertainty.

4.3 Comparative study of corruption models

Up to now, we considered only the Stain noise model to corrupt training data. In this comparative study we consider other noise models to confirm the relevance of our previous approach over other types of corruption that could have been considered. We provide here a comparison with three other synthetic noise models represented in Figure 6:

a- Gaussian noise. Corrupt by adding white noise applied uniformly over the entire image. For normalized intensities between 0 and 1, a corrupted pixel value x', corresponding to an initial pixel value x, is the realization of a random variable given by a normal distribution of mean x and variance σ^2 in the set: $[0.1, 0.2, 0.4, 0.8]$.

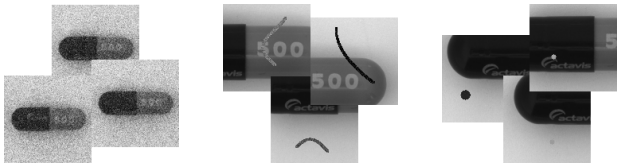
b- Scratch. Corrupt by adding one curve connecting two points whose coordinates are randomly chosen in the image and whose color is randomly selected in the gray scale range. The curve can follow a straight line, a sinusoidal wave or the path of a square root function.

c- Drops. Corrupt by adding 10 droplets whose color are randomly selected in the gray scale range and whose shape are circular with a random diameter (chosen between 1 and 2% of the smallest image dimension). The droplets partially overlap.

In addition, we have also considered the possibility to corrupt the training images with a combination of several models. We propose two hybrid models:

d- Mix1. This configuration corrupts training images with a combination of the Stain, Scratch and Drops models. We fix that 60% of the training images are corrupted with the Stain model while the remaining 40% are corrupted with the Scratch and Drops models in equal proportions.

e- Mix2. This configuration corrupts training images with a combination of the Stain and the Gaussian noise models. We fix that 60% of the training images are corrupted with the Stain model while the remaining 40% are corrupted with the Gaussian noise model.



(a) Gaussian noise. (b) Scratch. (c) Drops.

Fig. 6 Illustration of the Gaussian noise, Scratch and Drops models. The original clean image is the one presented in Figure 1.

Figure 7 allows to compare reconstructions obtained when the AESc network is trained over images corrupted with the newly introduced noise models with respect to the Stain noise model. First, these examples illustrate the convergence of the model towards the identity mapping when the Gaussian noise model is used as synthetic corruption. An analysis of the results obtained over the entire dataset reveals that the AESc + Gaussian noise model does almost not differ from the AESc network trained with unaltered images.

Compared to the the Gaussian noise, other models introduced before improve the identification of defective areas in the images. This is reflected by higher intensities of the reconstruction residual in the defective areas and close-to-zero reconstruction residual in the clean areas. With the exception of the Gaussian noise model, the Scratch model is the most conservative, among those considered, in the sense that most of the structures of the input images tend to be reconstructed identically. This

practice increases the number of false negative. Also, the Drops model restricts the structures detected as defective to sporadic spots. Finally, the three models based on the Stain noise (Stain, Mix1 and Mix2) provide the residuals that correlate the most with the segmentation mask.
Generally, models based on the Stain noise (Stain, Mix1 and Mix2) lead to the most relevant reconstruction for anomaly detection, i.e. lower residual intensities in clean areas and higher residual intensities in defective areas. More surprisingly, this statement remains true even if the actual defect looks more similar to the Scratch model than the Stain noise (Bottle sample in Figure 7). We recall that defects contained in the MVTec AD dataset are real observations of an anomaly. This reflects that models trained with synthetic corruption models that look similar to real ones do not necessarily generalize well to real defects.

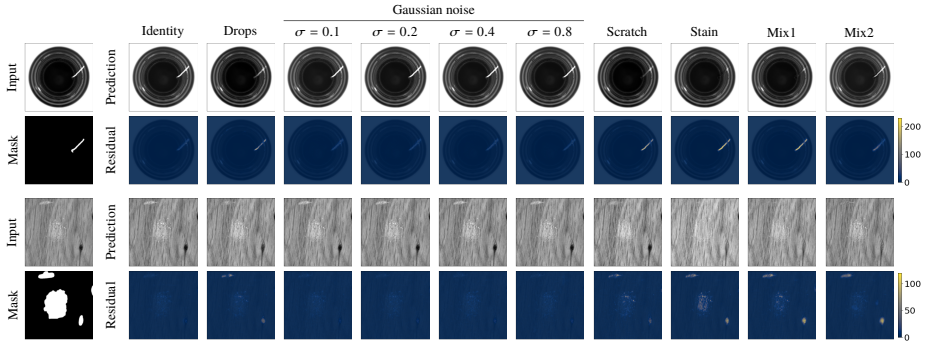


Fig. 7 Reconstructions obtained with the AESc network trained with different noise models. We consider here one defective object, namely a Bottle (first sample) and a defective texture, namely a Wood (second sample). Rows and columns are defined as in Figure 4.

Table 3 quantifies the impact of the synthetic noise model on the performances of the ASEc network to solve the image-wise detection task with a residual-based approach. The AESc + Stain configuration is the best performing in all use cases when considering the mean performances that are obtained over the entire dataset, as revealed by the previous qualitative study. The two hybrid models (Mix1 and Mix2) lead usually to slightly lower performances than those obtained with the Stain model. Those observations attest that the Stain model is superior to others and justify the choice of the Stain noise as our newly introduced approach to corrupt the training images with synthetic noise.

Table 3 Image-wise AUC obtained with the AESc network trained with different noise models with the residual-based problem formulation[a].

	Corruption	None	Drops	Gaussian noise (σ)				Scratch	Stain	Mix1	Mix2
				0.1	0.2	0.4	0.8				
Textures	Carpet	0.48	0.87	0.52	0.46	0.51	0.53	0.63	**0.89**	0.84	0.84
	Grid	0.52	0.94	0.55	0.69	0.59	0.72	0.79	**0.97**	0.91	0.96
	Leather	0.56	0.87	0.72	0.71	0.74	0.71	0.77	**0.89**	0.88	**0.89**
	Tile	0.88	0.94	0.94	0.92	0.90	0.92	0.95	**0.99**	0.98	0.96
	Wood	0.92	**0.99**	0.89	0.90	0.91	0.85	0.96	0.95	0.94	0.79
	Mean[b]	0.67	0.92	0.72	0.74	0.73	0.75	0.82	**0.94**	0.91	0.89
Objets	Bottle	0.77	**0.99**	0.82	0.85	0.81	0.75	0.91	0.98	0.98	0.97
	Cable	0.55	0.60	0.58	0.53	0.49	0.46	0.60	0.89	0.87	**0.90**
	Capsule	0.60	0.71	0.58	0.68	0.57	0.59	0.66	**0.74**	**0.74**	0.53
	Hazelnut	0.85	**0.98**	0.75	0.73	0.92	0.73	0.96	0.94	0.93	0.81
	Metal Nut	0.24	0.54	0.32	0.27	0.28	0.24	0.44	0.73	0.71	**0.86**
	Pill	0.70	0.79	0.69	0.71	0.73	0.68	0.78	**0.84**	0.77	0.78
	Screw	0.30	0.46	0.91	**0.99**	0.78	0.65	0.71	0.74	0.22	0.72
	Toothbrush	0.78	**1.00**	0.99	0.98	0.79	0.82	0.87	**1.00**	**1.00**	**1.00**
	Transistor	0.46	0.83	0.55	0.49	0.48	0.50	0.68	0.91	**0.92**	**0.92**
	Zipper	0.72	0.93	0.66	0.63	0.69	0.58	0.79	0.94	0.90	**0.98**
	Mean[c]	0.60	0.78	0.68	0.69	0.65	0.60	0.74	**0.87**	0.80	0.85
	Global mean[d]	0.62	0.83	0.70	0.70	0.68	0.65	0.77	**0.89**	0.84	0.86

[a] For each row, the best performing approach is highlighted in boldface and the second best is underlined.
[b] Mean AUC obtained over the classes of images belonging to the texture categories.
[c] Mean AUC obtained over the classes of images belonging to the object categories.
[d] Mean AUC obtained over the entire dataset.

5 Internal representation of defective images

The choice of an autoencoder network architecture, with or without skip connections, has been motivated by the willingness to compress the input image representation in order to regularize the reconstruction towards clean images only.
This section analyzes the latent representation of clean and defective images for both the AE + Stain and the AESc + Stain methods. The purpose of this study is to supplement the results presented in Section 4 by introducing a new problem formulation. A strong relation will be highlighted between our previous approach and this new study. In comparison to the initial *reconstruction-based* approach, this discussion addresses the anomaly detection task as the identification of *out-of-distribution* samples. In this formulation, each input image is represented in a another domain. This step has for purpose to produce tensors whose definition is particularly suited for this task. Then, a new metric quantifies the likelihood of a tensor to be sampled from the clean data distribution or not.

5.1 Formulating anomaly detection as an out-of-distribution sample detection problem

In this section, two fundamental notions are introduced in order to formulate the anomaly detection task as the identification of out-of-distribution samples. First, each sample is represented in every layer, by the layer feature map tensors. The originality of our formulation lies in the use of the output image of the network to represent each image as a tensor. As a reminder, this output is assumed to be a clean reconstruction of the arbitrary input image. Second, a measure allows to evaluate the distance of the input to the clean distribution, in each tensor space.

Definition of the tensor space. Let's denote an input image by $\mathbf{x}$ and the output of the network by $\hat{\mathbf{x}}$, with both $\mathbf{x}, \hat{\mathbf{x}} \in \mathbb{R}^{256\times256}$. Note that we fixed the image resolution to 256×256 for the MVTev AD dataset, but the equations can be adapted to handle any other image spatial dimension. We also define the operator $l_i(\cdot) : \mathbb{R}^{256\times256} \rightarrow \mathbb{R}^{n_i\times n_i\times c_i}$ which projects an input image to its activation tensor in the i^{th} layer. In this notation, n_i and c_i respectively denote the spatial dimension and the number of channels in layer i. As depicted in Figure 8, $l_1(\mathbf{x})$ corresponds to the activation tensor obtained after one convolutional layer while $l_6(\mathbf{x})$ is the activation tensor in the bottleneck.
For each layer of the encoder, a Δl_i tensor is computed for any input image $\mathbf{x}$ by subtracting the activation maps of $\hat{\mathbf{x}}$ from those of $\mathbf{x}$:

$$\Delta l_i(\mathbf{x}) = l_i(\mathbf{x}) - l_i(\hat{\mathbf{x}}) \,, \tag{2}$$

with $\hat{\mathbf{x}}$ being the closest clean reconstruction of $\mathbf{x}$, obtained by inferring $\mathbf{x}$ through the AE or AESc autoencoder. If $\mathbf{x}$ is sampled from the clean distribution, the images $\mathbf{x}$ and $\hat{\mathbf{x}}$ should be almost identical, as their activation tensors. This implies that the resulting Δl_i tensor of $\mathbf{x}$ should contains close-to-zero values. On the contrary, if $\mathbf{x}$ is sampled from the defective distribution, the corresponding Δl_i tensor should deviate from the zero tensor.

Definition of the distance to the clean distribution. It is expected that all the Δl_i tensors of a clean image contain close-to-zero values. However, it has been observed that, for a set of clean images, the components diverge more or less from this zero value depending on the input image. In order to take this phenomenon into account, we propose to represent each component of the Δl_i tensors, denoted by the index j, by the mean ($\mu_{i,j}$) and variance ($\sigma_{i,j}$) observed across the training set as described in Figure 9. The activation of the j^{th} component in each Δl_i is considered as abnormal (or out-of-distribution) if its value is not contained in the range $\mu_{i,j} \pm 3\sigma_{i,j}$.
Then, we quantify the level of anomaly affecting an image, by the number of out-of-distribution activations divided by the tensor dimension. This value is referred as the Out-of-Distribution Ratio (OoDR):

$$\mathrm{OoDR}_i(\mathbf{x}) = \sum_{j=1}^{n_i \times n_i \times c_i} \frac{\left[\Delta l_{i,j}(\mathbf{x})\right]_{\mathrm{OoD}}}{n_i \times n_i \times c_i} \tag{3}$$

$$\text{with} \begin{cases} \left[\Delta l_{i,j}(\mathbf{x})\right]_{\mathrm{OoD}} = 0, & \text{if } \mu_{i,j} - 3\sigma_{i,j} \leq \Delta l_{i,j}(\mathbf{x}) \leq \mu_{i,j} + 3\sigma_{i,j} \\ \left[\Delta l_{i,j}(\mathbf{x})\right]_{\mathrm{OoD}} = 1, & \text{else.} \end{cases}$$

The principal motivation behind this definition lies in a metric which scales well when dealing with high dimensional tensors.

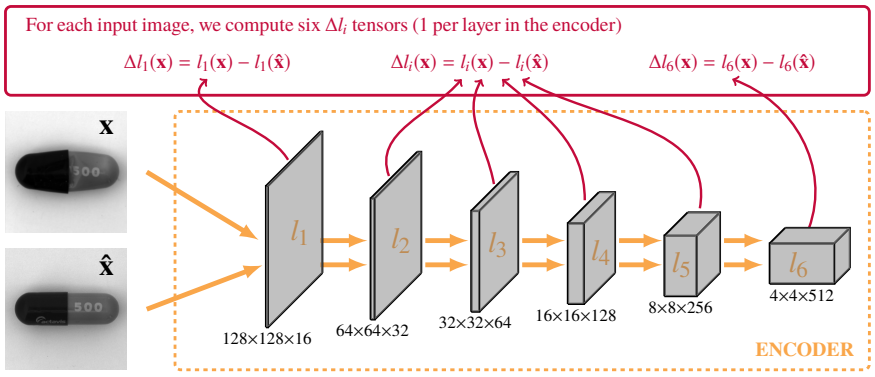
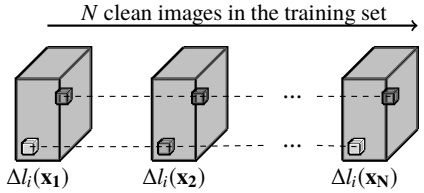


Fig. 8 For each input image $\mathbf{x}$, we infer both this image and its closest clean version $\hat{\mathbf{x}}$ obtained by passing $\mathbf{x}$ through AE or AESc. Hence, we obtain two activation tensors per layer i denoted by the operator $l_i(\cdot)$, one corresponding to $\mathbf{x}$ and the other to $\hat{\mathbf{x}}$. The six Δl_i tensors of an input image are obtained by computing the difference between the activation tensor of the input image and the ones of its reconstructed clean version.



For each component j in the Δl_i tensor with dimension $n_i \times n_i \times c_i$:

- $\mu_{i,j} = \sum_{n=1}^{N} \frac{\Delta l_{i,j}(\mathbf{x_n})}{N}$
- $\sigma_{i,j} = \sqrt{\frac{\sum_{n=1}^{N} \left(\Delta l_{i,j}(\mathbf{x_n}) - \mu_{i,j}\right)^2}{N}}$

Fig. 9 For each component j of the activation tensors, we compute the mean and the standard deviation of the neuron values across all clean images in the training set. Then, μ_j and σ_j represent the normal clean distribution of an activation value for a given component.

5.2 Out-of-distribution ratio for clean and defective samples

As introduced in Section 3, our networks are trained to minimize the reconstruction error over clean structures in the image. Despite training does not explicitly encourage the separation of clean and defective image tensors, we observe that this separation naturally emerges. This is revealed by the proximity/distance to zero of the OoDR_i distributions associated to clean/defective images. Moreover, the OoDR_i distributions appear to provide insightful justifications of the various anomaly detection performance obtained for different categories of images in Table 1. This analysis summarizes the results obtained over the image categories of the MVTec AD dataset by presenting three illustrative cases.

Case 1: Image category for which both AESc + Stain and AE + Stain achieve high performance. This first case focuses on the Bottle image category, for which both networks achieve close to 0.98 image-wise detection AUC (see Table 1). The OoDR_i distributions in both AE and AESc networks over the Bottle category are shown in Figure 10. As expected, we observe that the OoDR_i of clean images (blue) lead to smaller values than those of the corrupted/defective images (red). In the training set, the clean OoDR_i distributions are peaky in zero, while the corrupted ones are spread over a larger range of values. In the test set, the sharpness of the clean distributions is reduced which increases the overlap between the clean and the defective distributions.
With respect to the layer index, we observe that the OoDR_i distribution separation increases as we go deeper into the network, which is even more evident with the AESc architecture. Particularly, in the bottleneck ($i = 6$) of the AESc network, the two image distributions are almost perfectly separable.

Case 2: Image categories for which high performance is achieved with AESc + Stain but not with AE + Stain. This second case focuses on two texture image categories, Tile and Carpet. The AESc + Stain method achieves accurate detection of defective images (0.99 and 0.89 on Tile and Carpet, respectively) while image-wise detection AUC obtained with the AE + Stain methods are poor (0.57 and 0.43 on Tile and Carpet, respectively).
As shown in Figure 10, the clean (blue) and corrupted/defective (red) OoDR_i distributions of the AE + Stain method obtained over the Tile category follow the same trend. Moreover, the OoDR_i distributions of the clean images sampled from the test set are shifted to the right in comparison to those of the training set. Such trend is not visible with the AESc + Stain method, for which the OoDR_i distributions follow a similar behavior to the one observed for the Bottle dataset.
Regarding the Carpet category, the poor anomaly detection performance achieved by the AE + Stain model is also reflected by overlapping OoDR_i distributions. The OoDR_i distributions on test samples appear to be bi-modal and widely spread patterns. Again, those trends are not present with the AESc + Stain method where blue and red distributions appear to be more separable.

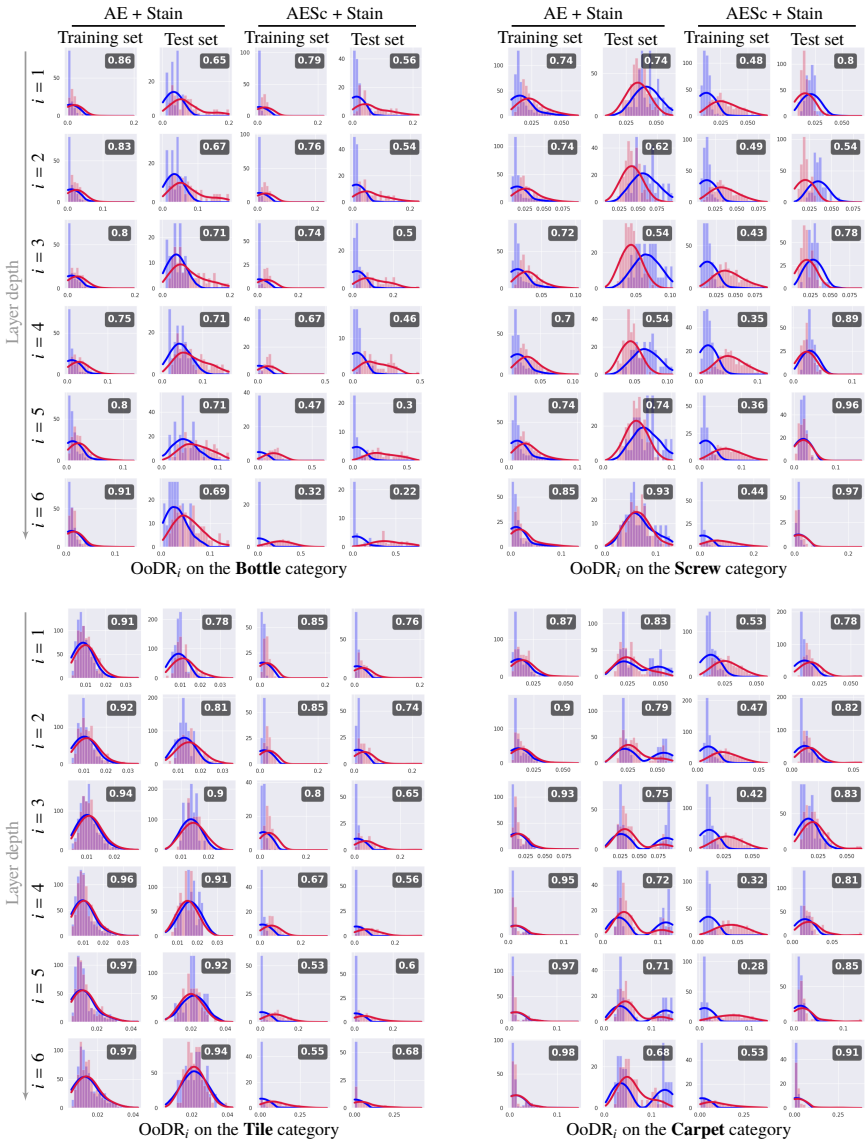


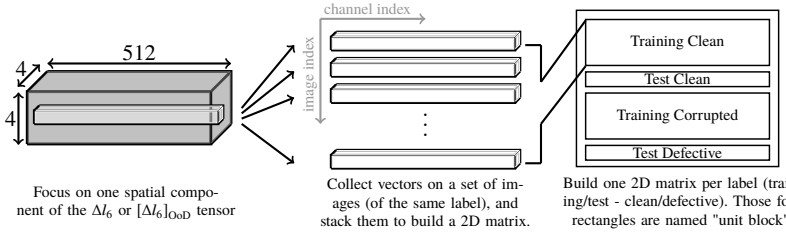
Fig. 10 Distribution of the OoDR$_i$ in each layer (only the encoder part) of the AESc and AE networks trained over images of the Bottle (up left), Screw (up right), Tile (down left) and Carpet (down right) categories and corrupted with the Stain noise model. For both architectures, the distribution for the training set (odd columns) and those for the test set (even columns) are presented apart. In each graph, OoDR$_i$ distributions related to clean images are represented in blue while OoDR$_i$ distributions related to corrupted (training)/defective (test) images are represented in red. The ratio of overlapping area between the two curves is provided in the gray rectangles (0 for non overlapping curves, 1 for identical curves).

Case 3: Image category for which poor performance is achieved both with AESc + Stain and AE + Stain compared to the state of the art. This third case focuses on the Screw category due to the poor image-wise detection AUC achieved (0.74 for AESc + Stain and 0.6 for AE + Stain) with respect to the perfect (1.00) detection obtained with the state of the art method. As shown in Figure 10, we observe an unexpected inversion of the clean and defective OoDR_i distributions over the test set for both AE and AESc networks. One possible explanation for this phenomenon is the variation of the luminosity between the clean images of the training set and the clean images of the test set. By carefully looking at the dataset, we have observed a dimming of the lightning in the clean image test set leading to a global change of the screws appearance characterized by less reflection and shading.

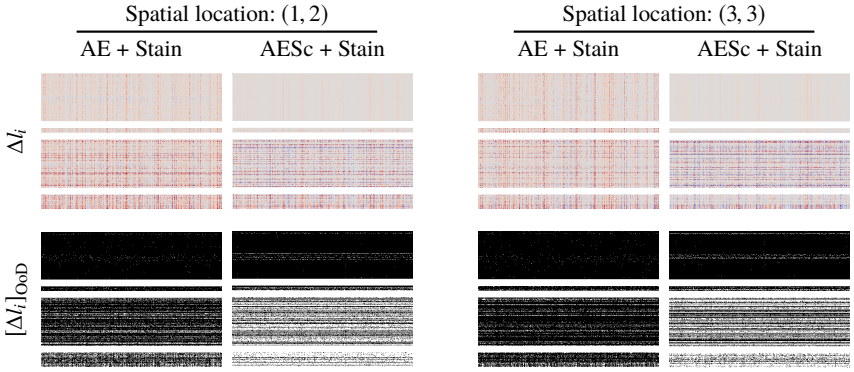
5.3 Analyzing out-of-distribution activations at the tensor component level

The objective of this section is to study whether some specific tensor components contribute more than others to the increase of the OoDR_i for defective images. This is of interest to potentially identify a subset of tensor components that are particularly discriminant to determine whether an image is clean or not.
In Section 5.2, we observed that the separation between the clean and defective OoDR_i distributions was more evident in the bottleneck than in the previous layers. For this reason, we focus the rest of our analysis on the samples separation achieved exclusively in the bottleneck.

In Figure 11, we provide a partial visualization of the Δl_6 and $[\Delta l_6]_{\text{OoD}}$ tensors computed in the bottleneck for both AE and AESc networks. The image category chosen for this experiment is the Bottle one, for which our method performs well. The tendency of the AESc + Stain model to produce near-to-zero Δl_6 tensors for clean images is strongly marked in this situation. Comparatively, the Δl_6 tensors obtained with AE + Stain for clean images are more noisy. This observation confirms the ability of the AESc + Stain model to separate better the clean and defective images than the AE + Stain model.
A more detailed analysis of this figure does not allow to state that some specific tensor components are particularly relevant to distinguish clean and defective images. If this were the case, vertical stripes would have been visible on the "unit blocks" obtained from the $[\Delta l_6]_{\text{OoD}}$ tensors.



(a) Legend describing how the dataset is summarized in one "unit block". Out of a Δl_6 or $[\Delta l_6]_{\text{OoD}}$ tensor of dimension $4 \times 4 \times 512$, only one one spatial resolution in considered by keeping only one $1 \times 1 \times 512$ vector per input image. This procedure is repeated for each image in the training clean/corrupted and the test clean/defective sets. All the vectors obtained from images belonging to the same class label are stacked together. Finally, one dataset is summarized by one "unit block" which is the ensemble of the training clean, test clean, training corrupted and test defective stacked vectors.



(b) According to the legend described above, the subfigure presents the "unit blocks" corresponding to the Bottle category. In the first row, the "unit blocks" are constructed from the Δl_6 tensors. The color map varies from the blue (negative activation values) to the red (positive activation values). In the second row, the "unit blocks" are constructed from the $[\Delta l_6]_{\text{OoD}}$ tensors. A black value indicates that the corresponding component j of the Δl_6 ranges between $\mu_{6,j} \pm 3\ \sigma_{6,j}$, while a white value stands for an out-of-distribution component.

Fig. 11 Visualization of the "unit blocks" obtained both from the Δl_6 and the $[\Delta l_6]_{\text{OoD}}$ tensors at two spatial locations in the bottleneck: (1, 2) and (3, 3). Those results are generated from the images of the Bottle category.

5.4 Using the out-of-distribution ratio as a measure quantifying the level of abnormality

Despite the fact that our networks are not explicitly trained to address the anomaly detection task as an out-of-distribution problem, it is possible to study the detection performances obtained when using the OoDR_i as a criterion to distinguish clean from defective images. Since we have observed previously that the best OoDR_i separation is achieved in the bottleneck, we carry out this analysis exclusively on

the OoDR_6 values. Results for all image categories are represented in Figure 12. In general, AESc + Stain achieves better image-wise detection AUC values than AE + Stain, which is consistent with previous observations. In comparison with the reconstruction-based approach (involving the computation of the $\mathcal{L}^2$-norm between an input image and its reconstruction), the out-of-distribution formulation achieves lower detection rates in general.

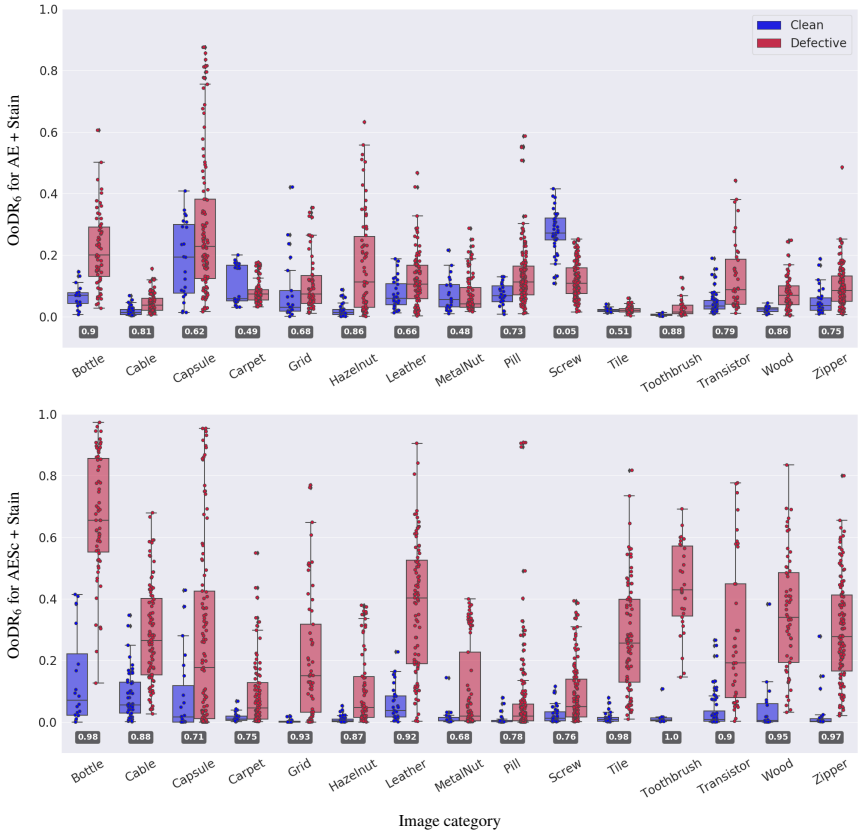


Fig. 12 Distribution of the OoDR_6 values obtained on the test clean (blue) and test defective (red) images on all image categories. Values below each box plot provide the image-wise detection AUC obtained when using the OoDR_6 as a decision rule. Upper graph provides the results obtained with the AE + Stain method and lower graph for the AESc + Stain method.

Intuitively, the design of a detector that would combine both the out-of-distribution and the reconstruction-based approaches into account could be used to enhance the detection of defective samples. Nevertheless, an improvement of the detection performance will be obtained only if the two decision criteria are complementary. Figure 13 however reveals that, in the current training configuration, the two criteria are strongly correlated. In this figure, the $\mathcal{L}^2$-norm is plotted as a function of the

OoDR_6 value. Generally, there is no decision boundary that performs significantly better than a vertical or an horizontal line. Hence, using a single criterion is sufficient, and there is no significant gain to expect from their combination.

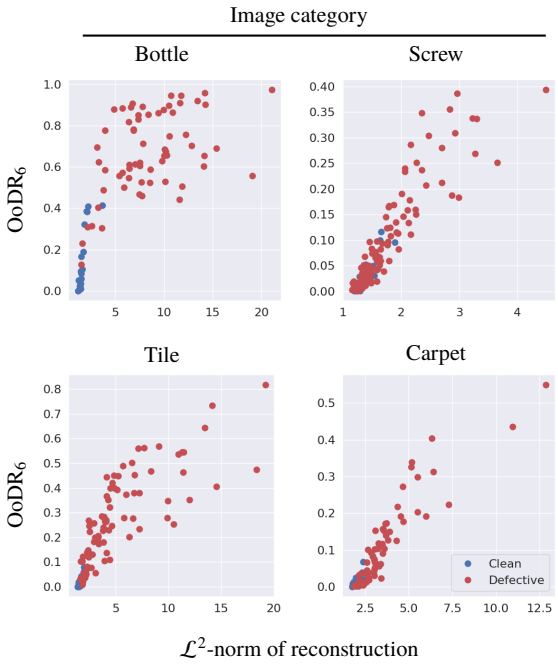


Fig. 13 For the test clean images (blue) and test defective images (red), we show the OoDR_6 value depending on the $\mathcal{L}^2$-norm of the reconstruction error. We performed this analysis for the four image categories addressed in Section 5.2 (Bottle, Screw, Tile and Capet).

5.5 Related works and perspectives

Multiple recent works have studied the internal representation of an image by a CNN to address anomaly detection [37, 38] or adversarial attacks detection [39, 40]. In the particular context of detecting adversarial samples, Granda et al. [40] revealed that only a limited subset of neurons is relevant to predict the class label. Hence, adversarial samples can be detected by observing how the state of those relevant neurons change compared to a class centroid. In comparison, since no class label is available during training in our case, our formulation relies on the comparison between the internal representations associated to the input and to its reconstructed clean image. It complements [40] by showing that, in absence of class supervision, all neurons of a layer are likely to change in presence of an anomaly.

Another recent work has considered the internal CNN representation for anomaly detecion, and has shown that constraining the latent representation of clean samples to be sparse improves the detection performance [41]. This is in line with our observations, since increased sparsity of the clean representation is also expected to favor a better separation of the OoDR_i distributions. This suggests that an interesting path for future research could consist in promoting OoDR_i separation explicitly during training.

6 Conclusion

Our work considers the detection of abnormal structure in an images based on the reconstruction of a clean version of this query image. Similarly to previous works, our framework builds on convolutional autoencoder and relies on the reconstruction residual or the prediction uncertainty, estimated with the Monte Carlo dropout technique, to detect anomalies. As an original contribution, we demonstrated the benefits of considering an autoencoder architecture equipped with skip connections, as long as the training images are corrupted with our Stain noise model to avoid convergence towards an identity operator. This new approach performs significantly better than traditional autoencoders to detect real defects on texture images of the MVTec AD dataset.
Furthermore, we also provided a fair comparison between the residual- and uncertainty-based detection strategies relying on our AESc + Stain model. Unlike the reconstruction residual, the uncertainty indicator is independent of the contrast between the defect and its surroundings, which is particularly relevant for low contrast defects localization. However, in comparison to the residual-based detection strategy, the uncertainty-based approach increases the false positive rate in the clean structures.

To better understand our evaluation, we conducted a throughout analysis of the internal representations of clean and defective samples. For this purpose, the anomaly detection task has been formulated as the identification of out-of-distribution samples. In other words, it identifies the samples for which the internal representations of the input and reconstructed image significantly differ, compared to an estimated normal variation level observed among clean samples. Our study revealed that some correlation exists between the performance of our initial approach and the natural trend of the network to separate clean from defective samples in this new problem formulation. This suggests that controlling explicitly the separation of clean and corrupted image representations during training could help in improving the anomaly detection performance on datasets where our approach is now ineffective.

7 Acknowledgments

Part of this work has been funded by the "Pôle Mecatech ADRIC" Walloon Region project, and by the Belgian National Fund for Scientific Research (FRS-FNRS).

References

[1] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. MVTec AD - A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9592–9600, 2019.

[2] Marco A.F. Pimentel, David A. Clifton, Lei Clifton, and Lionel Tarassenko. A review of novelty detection. *Signal Processing*, 99:215–249, 2014.

[3] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41:1–58, 2009.

[4] Nematullo Rahmatov, Anand Paul, Faisal Saeed, Won Hwa Hong, Hyun Cheol Seo, and Jeonghong Kim. Machine learning–based automated image processing for quality management in industrial Internet of Things. *International Journal of Distributed Sensor Networks (IJDSN)*, 15(10), 2019.

[5] Philipp Seebock, Jose Ignacio Orlando, Thomas Schlegl, Sebastian M. Waldstein, Hrvoje Bogunovic, Sophie Klimscha, Georg Langs, and Ursula Schmidt-Erfurth. Exploiting Epistemic Uncertainty of Anatomy Segmentation for Anomaly Detection in Retinal OCT. *IEEE Transactions on Medical Imaging*, 39(1):87–98, 2019.

[6] Chaoqing Huang, Jinkun Cao, Fei Ye, Maosen Li, Ya Zhang, and Cewu Lu. Inverse-Transform AutoEncoder for Anomaly Detection. *arXiv preprint arXiv:1911.10676*, 2019.

[7] Alex Kendall and Yarin Gal. What uncertainties do we need in Bayesian deep learning for computer vision? In *Proceedings of the Advances in Neural Information Processing Systems conference (NeurIPS)*, pages 5574–5584, 2017.

[8] Anne-Sophie Collin and Christophe de Vleeschouwer. Improved anomaly detection by training an autoencoder with skip connections on images corrupted with stain-shaped noise. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 2021.

[9] Diego Carrera, Giacomo Boracchi, Alessandro Foi, and Brendt Wohlberg. Detecting anomalous structures by convolutional sparse models. *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2015.

[10] Paolo Napoletano, Flavio Piccoli, and Raimondo Schettini. Anomaly detection in nanofibrous materials by CNN-based self-similarity. *Sensors*, 18(1):209, 2018.

[11] Benjamin Staar, Michael Lütjen, and Michael Freitag. Anomaly detection with convolutional neural networks for industrial surface inspection. *Procedia CIRP*, 79:484–489, 2019.

[12] Chong Zhou and Randy C. Paffenroth. Anomaly Detection with Robust Deep Autoencoders. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 665–674, 2017.

[13] Filippo Leveni, Milano Deib, Milano Deib, Milano Deib, and Milano Deib. PIF : Anomaly detection via preference embedding. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 2021.

[14] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss Functions for Image Restoration With Neural Networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57, 2016.

[15] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving Unsupervised Defect Segmentation by Applying Structural Similarity to Autoencoders. In *Proceedings of the International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, pages 372–380, 2019.

[16] Mohammad Sabokrou, Mohammad Khalooei, Mahmood Fathy, and Ehsan Adeli. Adversarially Learned One-Class Classifier for Novelty Detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3379–3388, 2018.

[17] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. In *Proceedings of the International conference on Information Processing in Medical Imaging (IPMI)*, pages 146–157, 2017.

[18] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. *Medical Image Analysis*, 54:30–44, 2019.

[19] Samet Akçay, Amir Atapour-Abarghouei, and Toby P. Breckon. Skip-GANomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection. *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2019.

[20] Christoph Baur, Benedikt Wiestler, Shadi Albarqouni, and Nassir Navab. Deep autoencoding models for unsupervised anomaly segmentation in brain MR images. In *Proceedings of the International MICCAI Brainlesion Workshop*, pages 161–169, 2018.

[21] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 1–16, 2016.

[22] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Proceedings of the Advances in Neural Information Processing Systems conference (NeurIPS)*, pages 2672–2680, 2014.

[23] Jake Snell, Karl Ridgeway, Renjie Liao, Brett D Roads, Michael C Mozer, and Richard S Zemel. Learning to generate images with perceptual similarity metrics. In *Proceedinngs of the IEEE International Conference on Image Processing (ICIP)*, pages 4277–4281, 2017.

[24] David Dehaene, Oriel Frigo, Sébastien Combrexelle, and Pierre Eline. Iterative energy-based projection on a normal data manifold for anomaly localization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[25] Matthias Haselmann, Dieter P. Gruber, and Paul Tabatabai. Anomaly Detection Using Deep Learning Based Image Completion. In *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1237–1242, 2018.

[26] Asim Munawar and Clement Creusot. Structural inpainting of road patches for anomaly detection. In *Proceedings of the IAPR International Conference on Machine Vision Applications (MVA)*, pages 41–44, 2015.

[27] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112, 2021.

[28] Shuang Mei, Yudan Wang, and Guojun Wen. Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model. *Sensors*, 18(4):1064, 2018.

[29] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing Normality to Detect Anomaly: Memory-Augmented Deep Autoencoder for Unsupervised Anomaly Detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1705–1714, 2019.

[30] Lu Wang, Dongkai Zhang, Jiahao Guo, and Yuexing Han. Image anomaly detection using normal data only by latent space resampling. *Applied Sciences*, 10(23):8660–8679, 2020.

[31] Aaron Van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Proceedings of the Advances in Neural Information Processing Systems conference (NeurIPS)*, number 30, pages 6306–6315, 2017.

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, 2015.

[33] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[34] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random Erasing Data Augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13001–13008, 2020.

[35] Ruth Fong and Andrea Vedaldi. Occlusions for effective data augmentation in image classification. *Proceedings of the International Conference on Computer Vision Workshop (ICCVW)*, pages 4158–4166, 2019.

[36] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting Chun Wang, Andrew Tao, and Bryan Catanzaro. Image Inpainting for Irregular Holes Using Partial Convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 85–100, 2018.

[37] Oliver Rippel, Patrick Mertens, and Dorit Merhof. Modeling the Distribution of Normal Data in Pre-Trained Deep Features for Anomaly Detection. *arXiv preprint arXiv:2005.14140*, 2020.

[38] Fabio Valerio Massoli, Fabrizio Falchi, Alperen Kantarci, Şeymanur Akti, Hazim Kemal Ekenel, and Giuseppe Amato. MOCCA: Multi-Layer One-Class Classification for Anomaly Detection. *arXiv preprint arXiv:2012.12111*, 2020.

[39] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. In *Proceedings of the Advances in Neural Information Processing Systems conference (NeurIPS)*, pages 7167–7177, 2018.

[40] Roger Granda, Tinne Tuytelaars, and Jose Oramas. Can the state of relevant neurons in a deep neural networks serve as indicators for detecting adversarial attacks? *arXiv preprint arXiv:2010.15974*, 2020.

[41] Davide Abati, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Latent space autoregression for novelty detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 481–490, 2019.