Unsupervised Variability Normalization For Anomaly Detection

Aitor Artola, Yannis Kolodziej, Jean-Michel Morel, Thibaud Ehret

UNSUPERVISED VARIABILITY NORMALIZATION FOR ANOMALY DETECTION

Aitor Artola[1] *Yannis Kolodziej*[2] *Jean-Michel Morel*[1] *Thibaud Ehret*[1]

[1]Université Paris-Saclay, CNRS, ENS Paris-Saclay, Centre Borelli, France
[2]Visionairy

ABSTRACT

Anomaly detectors are necessary to automatize industrial quality control. However, crafting such detectors is difficult due to the complexity and variability of the object even when working only with rigid objects. We show that adding a deep learning normalization step as a preprocessing step to model based detectors allows for better and more robust detections. This self-supervised normalization neural network is trained on non-anomalous data only. The proposed preprocessing method, followed by an automatic detector, achieves state-of-the-art results on rigid objects from the MvTec dataset.

Index Terms— Anomaly detection, deep-learning, denoising, self-similarity

1. INTRODUCTION

Historically, quality control in production lines has been done by human operators. Yet, automatizing the process can accelerate, reduce the production cost, and smooth out the performance variations caused by the operators' fatigue [1, 2].

As pointed out in several reviews [3], [4], detecting anomalies is not a classic classification problem [5], which requires well balanced and well defined classes. Anomalies do not form a class or even several classes. They are rare and have no definite pattern. The objects on which to detect the anomalies can also be extremely diverse. It is true that specific algorithms perform well for simpler cases like repetitive patterns in textile using Fourier representation [6] or when a perfect reference is available [7]. Yet, there is no general principle characterizing normality due to the complex nature of "normal" objects. An automatic self-supervised method working with a wide range of objects is faced with the variability of the appearance of normal objects, caused for example by changing light reflections or varying texture.

In this paper, we therefore propose to decompose the task into two steps (see Fig. 1), dealing separately with the two mentioned difficulties: the variability of the normal objects on the one side and the indefiniteness of anomalies on the other. To reduce the variability of normality, we first apply a self-supervised deep learning process that learns to create a normalized version of images of normal objects, where variations due to lighting, paint or texture are attenuated or eliminated. Once the data have been normalized by the neural network, we use a self-similarity based detection method. We shall see that this normalization process actually improves the detection of anomalies on rigid objects.

The paper is organized as follows. Section 2 discusses previous work. Section 3 presents a neural network learning a *variability normalization*. Section 4 details the statistically founded detector applied to the normalized images. Section 5 presents results of the proposed method on real data from MvTec and a comparison to other state-of-the-art anomaly detectors. We conclude in Section 6.

2. RELATED WORK

Anomaly detection is a basic industrial problem that is receiving much attention lately, especially after the publication of a new reference dataset by MvTec [8]. A recent review [4] classifies anomaly detector into five main categories:

Stochastic background models. The principle of these anomaly detection methods is that anomalies occur in the low probability regions of the background model [9, 10, 11].

Homogeneous background model. These methods estimate and generally subtract the background from the image to get a residual image representation on which detection is eventually performed [12, 13].

Local homogeneity models: center-surround detection. These methods are often used for creating saliency maps. Their rationale is that anomalies (or saliency) occur as local events contrasting with their surroundings [14, 15, 16]. The main limitation of this model is that a local anomaly is not necessarily a global anomaly.

Variational Sparsity-based background models. One recent non-parametric trend is to learn a sparse dictionary representing the background (i.e., *normality*) and to characterize outliers by their non-sparsity. For that each patch of the image is projected onto the dictionary and is detected as anomalous based on the quality of the reconstruction as well as the number of elements of the dictionary used for the reconstruction [17, 18, 19, 20].

Non-local self-similar background models. The basic assumption of this generic background model, applicable to most images, is that in normal data, each image patch belongs

Work supported by a CIFRE scholarship of the French Ministry for Higher Studies, Research and Innovation.

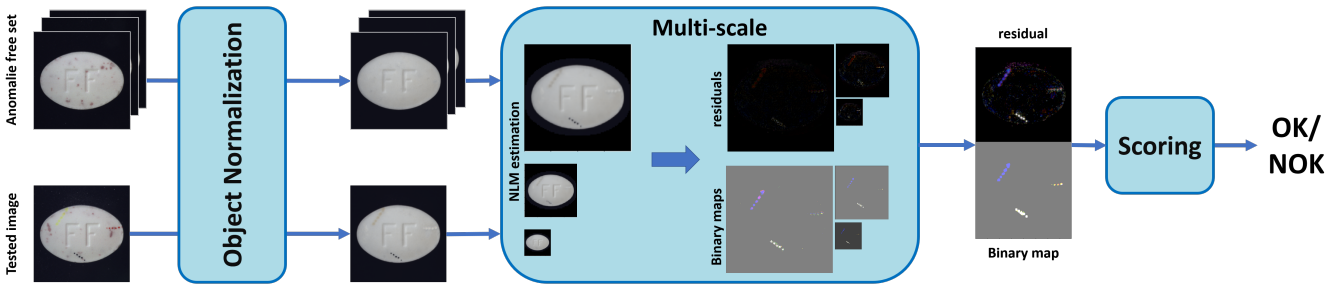

Fig. 1. Our proposed pipeline for robust anomaly detection in manufactured rigid objects.

Fig. 2. Example of variability of textures in the good pill class of the MvTech dataset [8]. Though the structure of the pills is the same (ellipsoids with big Fs engraved), the red spots are spread randomly and might be confused with anomalies.

Fig. 3. Effect of normalization (left: before, right: after) on the pill class of the MvTech dataset [8]. The normalization removes all "normal" red marks while keeping the anomalies.

to a dense cluster in the image's patch space. Anomalies instead occur far from their closest neighbors. This definition of an anomaly can be implemented by clustering the image patches or by a nearest neighbor search leading to a direct rarity measurement [21, 22].

More recently, sophisticated deep learning based methods have also been used to try to solve this problem. ITEA [23] proceeds on grey level images, normalizes their orientation and trains an auto-encoder on anomaly free images. AESC [24] is similar except that here the network is trained to denoise the image from a stain noise. SPADE [25] creates a feature base from a pre-trained neural network with the reference images and uses the L_2 norm between the features of an image and its kNN of the reference base as a score. It involves features at different layers to perform a multi-scale analysis. GANs have also been used to detect anomalies in medical images [26, 27].

3. PRE-PROCESSING: OBJECT NORMALIZATION

. Comparison to the state of the art in terms of AU-

The first step of our pipeline is the normalization of the data. We start by a classic geometric normalization of the object's position followed by a normalization of its color space, then proceed with a neural *normalization of the variability*.

Color and illumination normalization. Variations in exposure of the normal objects can result in a high color variability. The goal of the color normalization is to mitigate this high color variability. Let u_1 and u_2 two images with H_1 and H_2 their respective cumulative histograms. In order to

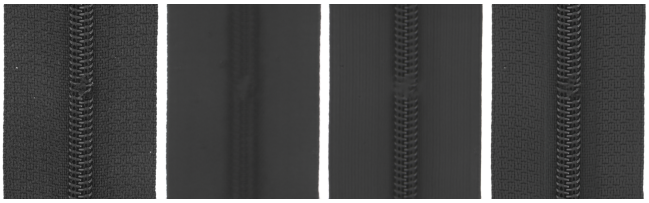

Fig. 4. Three sorts of normalization from rigid to flexible. From left to right: image without normalization, image after normalization using all whole images, using whole nearest images, finally using nearest patches.

bring the colors of two images in a common dynamics, Delon [28, 29] suggests to specify the cumulative histograms of both images onto the harmonic mean of the two cumulative histograms $H_{mid} = (\frac{1}{2}(H_1^{-1} + H_2^{-1}))^{-1}$. We extend this to N images (u_i) with respective cumulative histograms H_i by setting $H_{mid} = (\frac{1}{N}\sum_{i=1}^{N} H_i^{-1})^{-1}$.

Position normalization. Aligning correctly the objects allows for an easier comparison but also helps separating them from the background. We use here global affine alignment using the *inverse compositional* algorithm [30].

Variability normalization. Rigid objects can have conspicuous variable textures that can mask the real anomalies. This can be seen in Fig. 2. This variability must therefore be recognized as "normal" and eliminated before looking for anomalies. This variability is a "noise" that is different for each set. To avoid modeling it explicitly, we use the self-supervised Noise2noise [31] denoising network. This network f trains on pairs of realizations

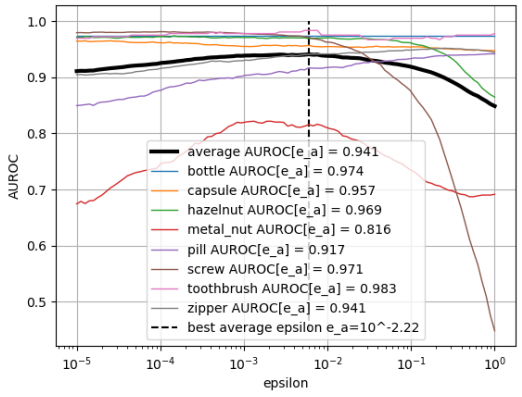

Fig. 5. Evolution of the score for the s_{ra} score with $\epsilon \in [10^{-5}, 1]$, the black curve is the average of AUROC of every objects at each ϵ

	Ours	SPADE [25]	ITEA [23]	AE [32]	AESC [24]
Bottle	97.4	97.2	94	**98**	**98**
Capsule	**95.7**	89.7	68	74	74
Hazelnut	**96.9**	88.1	86	90	94
Metal nut	**81.6**	71.0	67	57	73
Pill	**91.7**	80.1	79	76	84
Screw	97.1	66.7	**100**	68	74
Toothbrush	98.3	88.7	**100**	93	**100**
Zipper	94.1	**96.6**	80	90	94
Average	**94.1**	84.7	84.2	80.7	86.3

Table 1. Comparison to the state of the art in terms of AUROC.

of the same image (x_i^1, x_i^2), using the self-supervised loss $\mathcal{L} = \sum_{(x_i^1, x_i^2)} \|f(x_i^1) - x_i^2\|_1$. To create the pairs, we can pick any two images of our dataset of "normal" examples. For some examples, using any two images might be too drastic, particularly if the objects vary too much in shape. In such cases we use either only the nearest image neighbors, or the nearest patch neighbors, thus allowing for some deformation. As shown in Fig. 4, training the network with patches instead of whole images allows to keep fine detail and to keep the anomaly, while making the image more self-similar. Fig. 3 illustrates how the normalization removes the random red marks on the pills.

4. DETECTION USING SELF-SIMILARITY

Detection in a residual after background subtraction. For the final decision, we adapt the detection in the residual proposed in [21]. A "self-similar" background for each image is computed by a version of non-local means [33]. For each patch p of the query image u, the background is estimated as $\hat{p} = \frac{1}{N} \sum_{q \in \mathcal{Q}^K} q e^{-\frac{1}{2} \frac{\|p-q\|_2^2}{h^2}}$ where $N = \sum_{q \in \mathcal{Q}^K} e^{-\frac{1}{2} \frac{\|p-q\|_2^2}{h^2}}$, Q^K is the set of nearest neighbors and h the similarity parameter. The denoised patches $\hat{p}$s are then aggregated to create the model $\hat{u}$. The residual is given by $r(u) = u - \hat{u}$. It should contain only residual noise and the anomalies, characterized as the patches that show no similarity to other patches in the database. Once $r(u)$ has been estimated and a number of false alarms ϵ has been fixed, one can calculate the threshold γ_ϵ on the residual that, assuming $r(u)$ follows a centered normal law. This guarantees an upper bound on the number of false alarms, namely $\mathbb{P}\left(\left|\frac{r(u)-\mathbb{E}[r(u)]}{\sigma_{r(u)}}\right| \geq \gamma_\epsilon\right) = \frac{\epsilon}{N}$ *i.e.* $\gamma_\epsilon = \sqrt{2}\text{erfc}^{-1}\left(\frac{\epsilon}{N}\right)$.

	Bottle	Capsule	Metal nut	Pill	Screw	Toothbrush	Zipper	Leather	Wood
without	89.2	92.1	49.4	87.8	95.9	90.8	89.0	96.3	94.8
with	**97.4**	**95.7**	**81.6**	**91.7**	**97.1**	**98.3**	**94.1**	**99.8**	**99.9**

Table 2. Impact of the variability normalization on the AUROC. With the exception of the screw class, our variability normalization improves the detection by up to 30%.

Search for similar patches. The anomaly detector requires a search for similar patches in an anomaly-free database, which is a costly process. However, since the images have already been normalized in position thanks to the preprocessing, the search can be localized in a much smaller local region around the position of the reference patch. On top of that, we limit the search to the K images that are the most similar to the query image. This leads to a quick patch search. As a a backup strategy when no similar patch is found locally, we use partition trees to search more globally.

Multi-scale detection. The previous processing must be applied at multiple scales to capture all anomaly sizes. To combine the residuals r^ss, we define a multiscale residual r^{MS} by $r^{MS}(u) = \prod_{s \in \mathcal{S}} r^s(u)$. Multiplying the residuals allows us to inhibit detections that happens in a single scale while giving more weight otherwise. We also use binary activation maps b^s for each scale defined as $b^s(u, \epsilon) = 1$ if $f^s(u) >= \gamma_\epsilon^s$, 0 otherwise, where f^s is the normalized residual such that $f^s(u) = \left|\frac{r^s(u)-\mathbb{E}[r^s(u)]}{\sigma_{r^s(u)}}\right|$. The binary anomaly maps are merged by requesting detection at more than N_v scales. Thus we set $b^{MS}(u, \epsilon) = 1$ if $\sum_{s \in \mathcal{S}} b^s(u, \epsilon) \geq N_v$, 0 otherwise.

Scoring. From the binary anomaly maps, a normality score can be computed for each object. It is then sufficient to set a threshold to classify objects as abnormal. For that we propose s_{ra} that sums the absolute value of the residuals of the activated pixels in the binary map $s_{ra}(u, \epsilon) = \sum b^{MS}(u, \epsilon)|r^{MS}(u)|$. Using this metric, a small anomalous area with a large residual will be better represented.

5. EXPERIMENTS

Experimental settings. In this section, we compare our method to other state-of-the-art methods on the MvTech dataset [8]. As mentioned previously, our focus will be on the rigid objects of the dataset. We also show the impact of the proposed *variability normalization* pre-processing and detection maps that localize the detection for a few examples.

In all following experiments, we used $s = 3$ scales. The basic scale's dimension was in the range $[300, 512]$. It did not vary within the same object but from one to another depending on the computation time. The scaling factor η was 0.5 and the size of a patch 8×8. Candidates patches were searched in a 20×20 window around the position of the reference patch in the $K_s = 5 * (s + 1)$ closest images in the database. The examined patches overlap by half. The last parameter to fix is the NFA threshold ϵ. As mentioned in [4], this parameter implies a control on false positives: setting for example $\epsilon = 10^{-2}$ means that we should expect 1 false alarm every 100 images. Since there are about 200 test images per class, this means that the optimal ϵ should be about $10^{-2.3}$. We verified this hypothesis by looking at the evolution of the AUROC for ϵ in $[10^{-5}, 0]$. This is shown in Fig. 5 for each rigid object of the MvTech dataset as well as for the average. We can see that the value of ϵ that maximizes the average AUROC is $10^{-2.2}$. This is the value of ϵ that will be chosen for the rest of this section.

Comparison to state of the art. We compared our method with ITAE [23], AESC [24] and SPADE [25]. The proposed method achieves state-of-the-art results as shown as it can be seen in Table 1, it outperforms the state of the art by about 10% on average. Moreover, when our method does worse than another method, it only do so by less than 3%.

Impact of the variability normalization. We can see in Table 2 that this normalization improves the results in terms of AUROC for most classes (more than 30% for the metal nut class). This improvement is because the *variability normalization* makes the images easily comparable to the dataset. Since the images are easier to compare, the self-similarity based detector works better. We also show that this normalization might be helpful on some classes of the MvTec dataset (namely leather and wood) even though we only focus on rigid object in this paper.

Examples of anomaly segmentation. As seen in Section 4, the method computes binary anomaly maps b^{MS} used for the detection. These maps can also be visualized to localize the anomaly inside detected images. We show a few examples of these localization maps as well as some residuals in Fig. 6. The color of the binary map corresponds to the channel where the anomaly has been detected.

6. CONCLUSION

We devised an unsupervised anomaly detection method based on a *variability normalization* using a self-supervised neural network followed by an *a contrario* detection in the residual reconstruction. The variability normalization, which goal is to remove both the variability of the objects processed (*e.g.* random texture) and the variation in the acquisition process (*e.g.* different light reflections), is trained using a noise to noise [31] training process which is completely self-supervised and therefore can be done for each class of objects without modification. Since the method is completely self-supervised and only requires a few anomaly-free examples, the method is designed for rigid industrial object classes for which a moderately sized set of normal examples is available. We have also showed that, while being simple, this detection pipeline is both generic and powerful enough to outperform the current state of the art.

7. REFERENCES

[1] Hui-Chao Shang, You-Ping Chen, Wen-Yong Yu, and Zu-De Zhou, "Online auto-detection method and system of presswork quality," *The International Journal of Advanced Manufacturing Technology*, vol. 33, no. 7-8, pp. 756–765, 2007.

[2] Xiangqian Peng, Youping Chen, Wenyong Yu, Zude Zhou, and Guodong Sun, "An online defects inspection method for float glass fabrication based on machine vision," *The International Journal of Advanced Manufacturing Technology*, vol. 39, no. 11-12, pp. 1180–1189, 2008.

[3] Marco AF Pimentel, David A Clifton, Lei Clifton, and Lionel Tarassenko, "A review of novelty detection," *Signal Processing*, vol. 99, pp. 215–249, 2014.

[4] Thibaud Ehret, Axel Davy, Jean-Michel Morel, and Mauricio Delbracio, "Image anomalies: A review and synthesis of detection methods," *JMIV*, vol. 61, no. 5, pp. 710–743, 2019.

[5] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *IJCV*, vol. 115, no. 3, pp. 211–252, 2015.

[6] D-M Tsai and C-Y Hsieh, "Automated surface inspection for directional textures," *Image and vision computing*, vol. 18, no. 1, pp. 49–62, 1999.

[7] Pin Xie and Sheng-Uei Guan, "A golden-template self-generating method for patterned wafer inspection," *Machine Vision and Applications*, vol. 12, no. 3, pp. 149–156, 2000.

[8] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger, "Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection," in *CVPR*, 2019.

[9] Bo Du and Liangpei Zhang, "Random-selection-based anomaly detector for hyperspectral imagery," *IEEE Transactions on Geoscience and Remote sensing*, vol. 49, no. 5, pp. 1578–1589, 2011.

[10] Xianghua Xie and Majid Mirmehdi, "Texems: Texture exemplars for defect detection on random textured surfaces," *IEEE PAMI*, vol. 29, no. 8, pp. 1454–1464, 2007.

[11] Lionel Tarassenko, Paul Hayton, Nicholas Cerneaz, and Michael Brady, "Novelty detection for the identification of masses in mammograms," 1995.

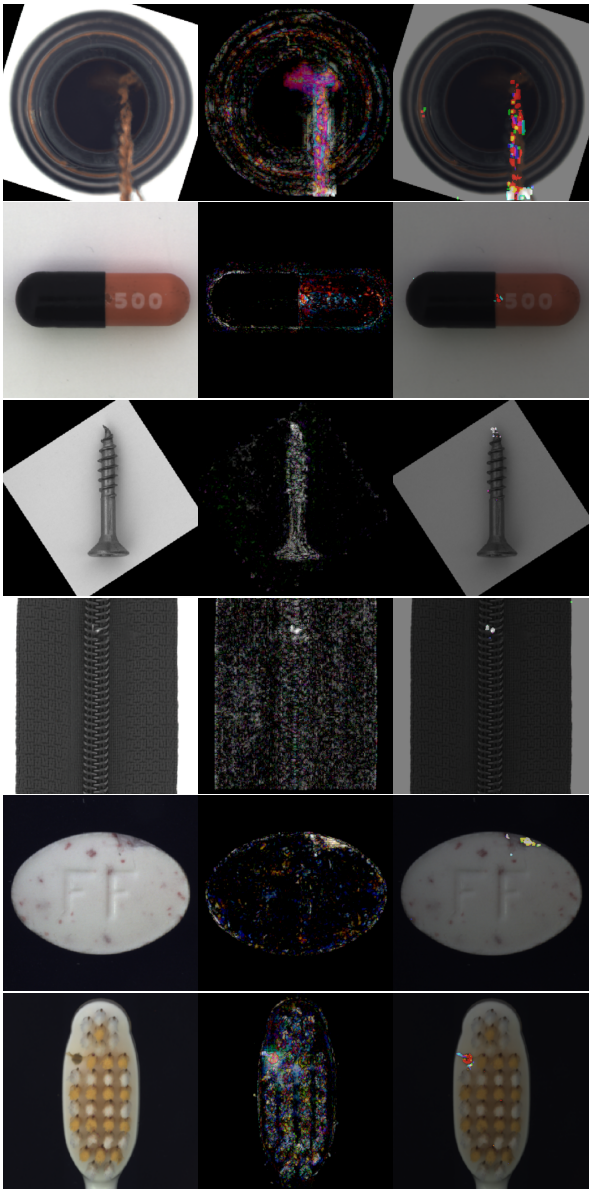

Fig. 6. Examples of anomaly segmentation on anomalous images. For each object, from left to right: the input, the logarithm of the multiscale residual, the multi-scale binary detection map overlaid on the input.

[12] Du-Ming Tsai and Tse-Yun Huang, "Automated surface inspection for statistical textures," *Image and Vision computing*, vol. 21, no. 4, pp. 307–323, 2003.

[13] Jinwon An, "Variational Autoencoder based Anomaly Detection using Reconstruction Probability," *Arxiv*, 2016.

[14] Laurent Itti, Christof Koch, and Ernst Niebur, "A model of saliency-based visual attention for rapid scene analysis," *PAMI*, vol. 20, no. 11, pp. 1254–1259, 1998.

[15] Toshifumi Honda and Shree K Nayar, "Finding" anomalies" in an arbitrary image," in *ICCV*. IEEE, 2001.

[16] Irving S Reed and Xiaoli Yu, "Adaptive multiple-band cfar detection of an optical pattern with unknown spectral distribution," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 10, pp. 1760–1770, 1990.

[17] Ran Margolin, Ayellet Tal, and Lihi Zelnik-Manor, "What makes a patch distinct?," in *CVPR*, 2013.

[18] Giacomo Boracchi, Diego Carrera, and Brendt Wohlberg, "Novelty detection in images by sparse representations," in *2014 IEEE Symposium on Intelligent Embedded Systems*. IEEE, 2014, pp. 47–54.

[19] Diego Carrera, Giacomo Boracchi, Alessandro Foi, and Brendt Wohlberg, "Detecting anomalous structures by convolutional sparse models," in *2015 International Joint Conference on Neural Networks*. IEEE, 2015, pp. 1–8.

[20] Diego Carrera, Giacomo Boracchi, Alessandro Foi, and Brendt Wohlberg, "Scale-invariant anomaly detection with multiscale group-sparse models," in *ICIP*. IEEE, 2016.

[21] Axel Davy, Thibaud Ehret, Jean-Michel Morel, and Mauricio Delbracio, "Reducing anomaly detection in images to detection in noise," in *ICIP*. IEEE, 2018.

[22] Maria Zontak and Israel Cohen, "Defect detection in patterned wafers using anisotropic kernels," *Machine Vision and Applications*, vol. 21, no. 2, pp. 129–141, 2010.

[23] Ye Fei, Chaoqin Huang, Cao Jinkun, Maosen Li, Ya Zhang, and Cewu Lu, "Attribute restoration framework for anomaly detection," *IEEE Transactions on Multimedia*, 2020.

[24] Anne-Sophie Collin and Christophe De Vleeschouwer, "Improved anomaly detection by training an autoencoder with skip connections on images corrupted with stain-shaped noise," *arXiv*, 2020.

[25] Niv Cohen and Yedid Hoshen, "Transformer-based anomaly segmentation," *arXiv*, 2020.

[26] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs, "Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery," *Arxiv*, 2017.

[27] Md Mahfuzur Rahman Siddiquee, Zongwei Zhou, Nima Tajbakhsh, Ruibin Feng, Michael B Gotway, Yoshua Bengio, and Jianming Liang, "Learning fixed points in generative adversarial networks: From image-to-image translation to disease detection and localization," in *ICCV*, 2019, pp. 191–200.

[28] Julie Delon, "Midway image equalization," *Journal of Mathematical Imaging and Vision*, vol. 21, no. 2, pp. 119–134, 2004.

[29] Thierry Guillemot and Julie Delon, "Implementation of the Midway Image Equalization," *Image Processing On Line*, vol. 6, pp. 114–129, 2016.

[30] Simon Baker and Iain Matthews, "Equivalence and efficiency of image alignment algorithms," in *CVPR*. IEEE, 2001.

[31] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila, "Noise2noise: Learning image restoration without clean data," in *ICML*, 2018.

[32] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger, "Improving unsupervised defect segmentation by applying structural similarity to autoencoders," *Proceedings of the 14th International Joint Conference on*

Computer Vision, Imaging and Computer Graphics Theory and Applications, 2019.

[33] Antoni Buades, Bartomeu Coll, and J-M Morel, "A non-local algorithm for image denoising," in *CVPR*. IEEE, 2005.