Autoencoder-based anomaly detection for surface defect inspection

Du-Ming Tsai [*], Po-Hao Jen

Department of Industrial Engineering & Management, Yuan-Ze University, 135 Yuan-Tung Road, Chung-Li, Taiwan

ARTICLE INFO

ABSTRACT

In this paper, the unsupervised autoencoder learning for automated defect detection in manufacturing is evaluated, where only the defect-free samples are required for the model training. The loss function of a Convolutional Autoencoder (CAE) model only aims at minimizing the reconstruction errors, and makes the representative features widely spread. The proposed CAE in this study incorporates a regularization that improves the feature distribution of defect-free samples within a tight range. It makes the representative feature vectors of all training samples as close as possible to the mean feature vector so that a defect sample in the evaluation stage can generate a distinct distance from the trained center of defect-free samples. The proposed CAE model with regularizations has been tested on a variety of material surfaces, including textural and patterned surfaces in images. The experimental results reveal that the proposed CAE with regularizations significantly outperforms the conventional CAE for defect detection applications in the industry.

1. Introduction

Machine vision is an effective non-contact technology for automated defect inspection in the manufacturing process. Most of the traditional machine vision techniques are based on texture analysis. A set of discriminative features are extracted from the spatial or the spectral domain of the test image. A high-level multiple dimensional classifier such as Support Vector Machine (SVM) or Random Forest is then applied to identify defect samples. The success of the classification highly relies on the human experts to extract and select representative features based on the local gray-level (or color) and structure variations of a defect in the test image.

In the manufacturing environment, it is quite easy to collect normal samples as many as required. However, it is difficult to collect a sufficient number of defective samples in a short period of time to train robust classification models for defect detection. The machine vision methods currently available need handcrafted features based on the characteristics of individual defect types of a specific product, where the defect samples may not be sufficient for the analysis.

In this paper, the deep learning technique is explored to tackle the defect detection task without defect samples for the training. The proposed method is image-wise defect detection, i.e. it classifies a test image as defective or defect-free. It is not used for pixel-wise defect segmentation. The unsupervised convolutional autoencoder (CAE) is applied to extract the representative features that can well describe the distribution from a set of normal samples. The loss function of the conventional CAE measures only the reconstruction errors. It could make the extracted feature values widely spread in the high-dimensional variable space. When the trained CAE is used for anomaly detection, the encoded features of a defect sample image may thus fall within the range of the normal samples' variable space. It could make the extracted features indistinguishable between normality and abnormality. A regularization penalty is thus included in the original CAE loss function to limit the spread of the learned feature values for normal training samples. It is expected that the distances of feature vectors between the normal samples are close to each other and the unseen defect samples yield sufficiently large distances from the normal samples. The proposed CAE-based model, denoted by λ-CAE, is tested on various material surfaces for defect detection, including textural surfaces and patterned surfaces. The proposed method is also compared to autoencoder-variant models based on encoded features and image reconstruction error.

This paper is organized as follows. Section 2 reviews the related work on defect detection with traditional machine vision and deep learning techniques. Anomaly detection with autoencoders for various applications is also discussed. Section 3 presents the original CAE and the regularized λ-CAE models for defect detection. Section 4 discusses the experimental results on various material surfaces. The paper is concluded in Section 5.

---

* Corresponding author.
E-mail address: iedmtsai@saturn.yzu.edu.tw (D.-M. Tsai).

2. Related work

Traditional machine vision methods rely on the extraction of hand-crafted features or the design of discrimination measures to identify defects in a test image. Texture analysis techniques [1] in image processing have been popularly used for defect detection in various material surfaces. Local texture features or descriptors are extracted from the spatial or the spectral domain [2] of a test image. Discriminant classifiers are then applied to identify defects. The statistical features or texture features based on local variance [3], local binary pattern [4] and co-occurrence matrices [5,6] are the commonly used spatial approach for surface defect detection. The spectral features based on Fourier transforms [7,8], Gabor transforms [9,10] and wavelet transforms [11,12] are the typical approach for defect detection in textured surfaces. The local feature extraction techniques only work well for uniform and homogeneously textured surfaces. They must be carefully designed for individual material surfaces. They could fail for products that contain non-periodically patterned surfaces.

Traditional machine learning techniques have been applied for anomaly detection. Ahmed et al. [13] reviewed various machine learning and network techniques for anomaly detection, including Support Vector Machine (SVM), Gaussian mixture models, Bayesian networks, clustering, etc. Domingues et al. [14] also evaluated a variety of machine learning algorithms for outlier detection, including probability-based, distance-based, neighbor-based, information theory-based and neural network-based models. Mohammadi-Ghazi et al. [15] used a boosted conditional Gaussian mixture models for novelty detection of sensor network data. Sadooghi and Khadem [16] proposed a one-class SVM using nonlinear features for novelty detection of bearing vibration data. Chakraborty et al. [17] used stacked autoencoders and probabilistic neural networks for outlier detection of vector-valued data.

The end-to-end deep learning [18] has become the dominated technique in the computer vision community for a wide variety of applications. A deep-learning neural network can automatically carry out feature extraction and classification tasks in one model with back-propagation optimization from a large set of training samples. Wang et al. [19] reviewed various deep learning techniques for smart manufacturing applications. Convolutional Neural Networks (CNNs, [20]) are the most popular deep learning models applied to the manufacturing industry. Weimer et al. [21] evaluated CNN architectures and their hyperparameter settings for automated industrial inspection. CNNs have been applied to the detection of cracks on reactors [22], steel surfaces [23], and liquid crystal display (LCD) panels [24]. Dai et al. [25] proposed a deep learning scheme for soldering defect detection in printed-circuit-boards (PCBs). It first applies the YOLO object detector to localize solder joints. Active learning is then used to reduce labeling work and generate a good classification model. Wang et al. [26] used the faster R-CNN object detector to identify defects in complex product images. It requires the manual annotation of bounding boxes and classes for each training image. Niu et al. [27] used the Generative Adversarial Network (GAN) to create synthesized defect images, and then applied the VGG models for defect detection and defect classification. The current CNN approach for automated surface defect detection generally requires a huge amount of both defect-free and defective image samples for the modeling training.

In the case where the defect image samples are rare or not available, the unsupervised learning is applied for anomaly detection. Given a large set of defect-free training samples, the goal of anomaly detection is to learn a feature representation that can abstractly describe the distribution or inherent pattern of the defect-free samples. For an inspection image, the trained model is then used to extract the representative features, or to reconstruct the image. The anomaly that deviates from the normal samples can be identified by measuring the posterior probability of the extracted features or the approximation error (residual) of the reconstructed image.

Kwon et al. [28] reviewed deep learning-based networks for anomaly detection, including Restricted Boltzmann Machine, deep belief network, deep neural network, recurrent neural network and autoencoders. Ruff et al. [29] and Ruff et al. [30] proposed Deep SVDD (Support Vector Data Description) for unsupervised anomaly detection. The objective of Deep-SVDD is inspired by the one-class SVM. It learns a convolutional neural network that minimizes the volume of a data-enclosing hypersphere. Points mapped outside the sphere are identified as anomalies.

The autoencoders (AEs) or generative models [31] are typical unsupervised neural networks for anomaly detection. For probabilistic models, the feature vector V_x of a test image x is extracted from the trained model. An anomalous sample is the one with low probability, i.e.

$$p_\Theta(V_x) \leqslant T_p \tag{1}$$

where Θ are the parameters of a neural network model learned from the defect-free samples, and T_p is the user-specified threshold.

For image reconstruction models, the test sample x is reconstructed from a trained generative model and the reconstruction residual is calculated by

$$\Delta\varepsilon_x = \|x - G_\Theta(x)\| \tag{2}$$

where G_Θ is the trained generative model with parameters Θ from a set of defect-free samples. If $\Delta\varepsilon_x \geqslant T_{\Delta\varepsilon}$ for some user-specified threshold $T_{\Delta\varepsilon}$, the test image x is an anomaly.

For the AE-variant approach, Makhzani and Frey [32] proposed a deep Convolutional Autoencoder (DCAE) to learn shift-invariant sparse representations for anomaly detection. Chow et al. [33] used convolutional autoencoders to detect defects on concrete structures. Pixel-wise defects are detected from the difference between the input image and its reconstructed image from the autoencoder model. It works well for defects with significantly large sizes in non-textured surfaces. An and Cho [34] trained the Variational Autoencoder (VAE) model from a set of normal samples. The reconstruction probability of a test image from the trained VAE is used to measure the anomaly. Zhou and Paffenroth [35] proposed a Robust deep autoencoder inspired by the Robust PCA. It splits the input into a clear reconstructed signal and the outlier/noise. The training of Robust AEs is unsupervised. However, both unlabeled defect-free and defective samples must be provided for the training. Yang et al. [36] proposed a fully convolutional autoencoder for defect detection in textured surfaces. It calculated the residual image by subtracting the autoencoder-generated texture background from the input image. A dual-thresholding is then applied to segment defects in the residual image. It works very well for defect segmentation in structural texture surfaces, such as textile fabrics.

For the GAN-variant approach, Schlegl et al. [37,38] proposed a deep convolutional GAN network, named AnoGAN, to learn a manifold of normal samples for anomaly detection. The anomaly score of a test image is measure by the residual loss and discrimination loss from the trained AnoGAN model. Zenati et al. [39] trained a GAN to fit the distribution of normal samples. It learns a generator and a discriminator. The reconstruction error plus the discriminator's confidence is used to measure the anomaly. Deecke et al. [40] also used GAN for anomaly detection by finding a good representation in the latent space of the generator. It needs on-line training of the GAN model for each test image. The loss function value of the GAN model is used as the anomaly measure.

The current deep learning methods for anomaly detection are based on AE-variant or GAN-variant models. GAN training is generally difficult in practice due to the vanishing gradient in high-dimensional spaces, as noted by Arjovsky and Bottou [41]. The VAE is also a generative model. It is relatively difficult to train [42], compared to the straightforward CAE. The discriminant measure based on the mean reconstruction error from autoencoders could be insignificant when the defect size is relatively very small w.r.t. the input image size. (Please refer to Appendix A for the performance comparison between CAE and VAE, and Appendix B
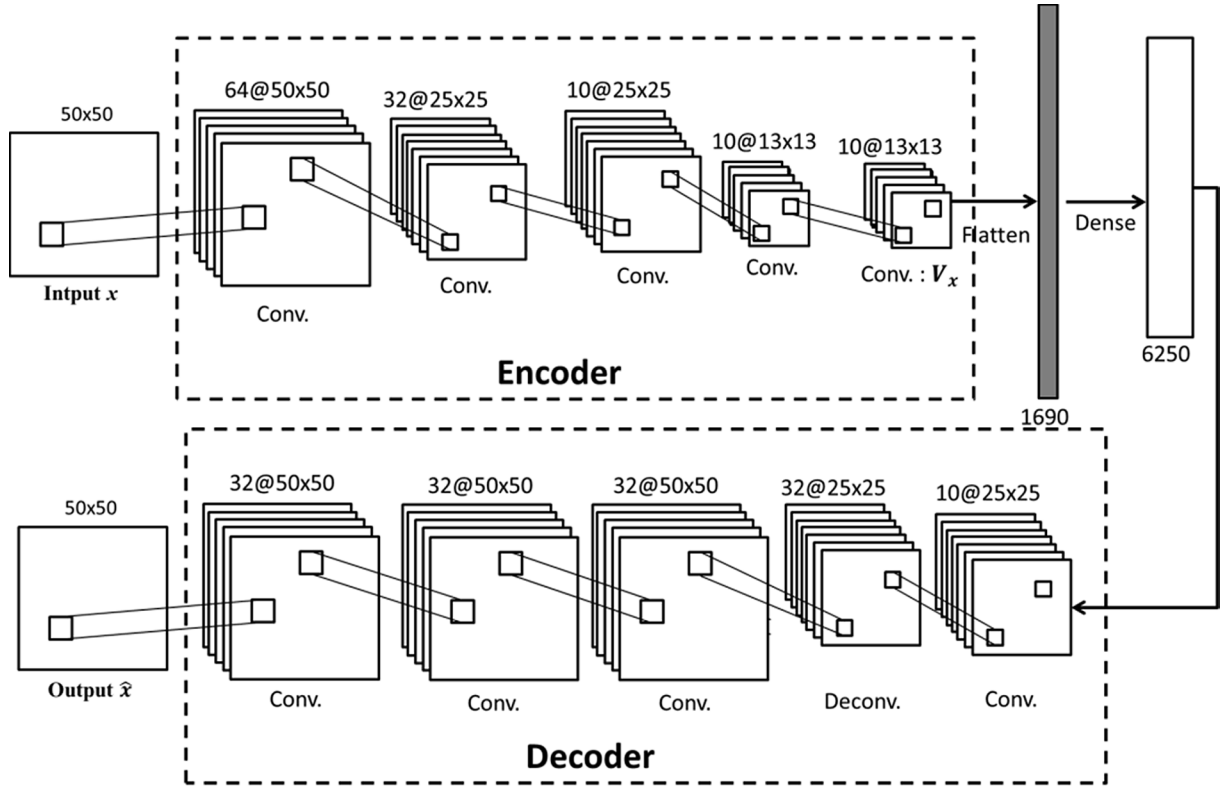
Fig. 1. Proposed CAE architecture for anomaly detection.

for the comparison between encoded features and image reconstruction errors.) In this study, regularized CAE models are thus proposed to extract representative features from a set of normal sample images. The proposed CAE models are trained so that the reconstruction error is minimized while the variation of the extracted feature vectors from all defect-free training samples are as small as possible. Simple thresholding based on the SPC (Statistical Process Control) is applied to distinguish the distance between the test image's feature vector and the mean feature vector of defect-free training images for defect-image identification. The proposed method does not require defective samples for the model training, and does not require handcrafted features for the discrimination. It is especially well suited for new product quality inspection in manufacturing, where the defect samples are not available or very rare to collect.

3. AE models for anomaly detection

3.1. CAE model

The autoencoder model used in this study is based on Convolutional Autoencoder (CAE) for practical implementation in manufacturing. The autoencoder architecture for unsupervised learning is composed of two parts, encoder and decoder. The encoder takes the raw image as the input, and the abstract representation from the encoder is then the input to the decoder. The encoder involves a series of convolutional layers and downsampling to compress the original data in a high-dimensional space into an abstract in a lower-dimensional space. It is expected the abstract representation can well describe the distribution of the defect-free training samples. The decoder is a generative model that involves a series of deconvolution and upsampling to reconstruct the image from the representative features.

Let F and G be the encoder and decoder, then

$$F : R^n \to R^m, \mathrm{x} \to \mathrm{F}(\mathrm{x}) \tag{3}$$

$$G : R^m \to R^n, \mathrm{F}(\mathrm{x}) \to \mathrm{G}(\mathrm{F}(\mathrm{x})) \tag{4}$$

where

n is the dimension of the input x;
m is the dimension of the abstract representation, and $m << n$.

The autoencoder is trained so that the reconstruction error is minimized:

$$\underset{\varphi,\theta}{Min}\mathrm{L}(\boldsymbol{x}, \mathrm{G}_\theta(\mathrm{F}_\varphi(\boldsymbol{x}))) \tag{5}$$

where φ and θ are the parameters to learn for the encoder and decoder, respectively. The CAE architecture used in this study for defect detection is depicted in Fig. 1. It consists of 5 convolutional layers in the encoder, and no pooling layers are applied. The decoder contains 6 deconvolutional/convolutional layers.

The loss function of the conventional CAE is measured by the mean square error (MSE), i.e.

$$\underset{\varphi,\theta}{Min}L(\mathrm{X}) = \frac{1}{N}\sum_{i=1}^{N}\left\|\mathrm{x}_i - \widehat{x}_i\right\|^2 \tag{6}$$

where

$\widehat{x}_i$ is the reconstructed image of the input image x_i, and $\widehat{x}_i = \mathrm{G}_\theta(\mathrm{F}_\varphi(\mathrm{x}_i))$;
$\mathrm{X} = \{\mathrm{x}_1, \mathrm{x}_2, ..., \mathrm{x}_N\}$, a set of defect-free samples for the model training.

The feature maps in the last convolutional layer of the encoder are used to form the feature vector that describes the inherent properties of defect-free sample images. Let w_k be the k-th feature map of size $r \times c$, and $k = 1, 2, ..., K$ (i.e. a total of K feature maps). The 2D feature map w_k is converted into a 1D vector v_k of size $1 \times r \cdot c$. The K hidden

representative vectors $v'_k s$ are concatenated to form a large vector V of size $1 \times r \cdot c \cdot K$. The concatenated vector V contains the final representative features used for anomaly detection.

Let $\overline{V} = \frac{1}{N}\sum_i V_i$ be the mean feature vector of all defect-free training samples in X, and V_i the feature vector of sample x_i. The distance of a test image I with representative feature vector V_I is given by

$$d_I = \left\| V_I - \overline{V} \right\| \tag{7}$$

It is expected that a defect-free test image represented by the extracted feature vector V is very close to the training samples, whereas a defective test image is far away from the training samples in the feature vector space.

3.2. Improved loss functions of CAE

CAE is a relatively simple model, compared to VAE, for the training. It is also more efficient in the evaluation time. The preliminary experiment shows that the extracted features from CAE is more distinctive for defect detection when only defect-free sample images are used for the model training. (See Appendix A for the comparison of CAE and VAE.)

As noted in Eq. (6), the loss function of the conventional CAE simply measures the reconstruction error $\|x - \hat{x}\|^2$ during the backpropagation process. The resulting feature maps trained in the encoder could be widely varied for individual defect-free samples. In order to generate the feature vectors that are tightly close to each other during the CAE model training, regularization is introduced to the original loss function. The proposed regularization tends to minimize the distance of each defect-free sample x_i in the training set X to the mean feature vector $\overline{V}$. That is

$$d_i = \left\| V_i - \overline{V} \right\| \tag{8}$$

In each epoch of the model training, the mean feature vector $\overline{V}$ is not available until all training samples in X are evaluated. The mean feature vector from the previous epoch is thus used as the center of all training samples. The proposed new loss function of CAE at epoch l is given by

$$L_V^{(l)}(X) = (1 - \lambda)\frac{1}{N}\sum_{i=1}^{N}\left\| x_i - \hat{x}_i^{(l)} \right\|^2 + \lambda\frac{1}{N}\sum_{i=1}^{N}\left\| V_i^{(l)} - \overline{V}^{(l-1)} \right\|^2 \tag{9}$$

where

$\overline{V}^{(l-1)} = \frac{1}{N}\sum_{i=1}^{N}V_i^{(l-1)}$, the mean feature vector at epoch l-1, and $\overline{V}^{(0)} = \mathbf{0}$;
λ is the penalty factor, and $0 \leqslant \lambda \leqslant 1$.

When $\lambda = 0$, it is the conventional CAE. When $\lambda > 0$, the regularization is put into action. The first term of the loss function L_V minimizes the reconstruction error with the weight $(1 - \lambda)$. The second term makes the feature vector V_i of each training sample x_i as close as possible to its center $\overline{V}$. The regularization term can be interpreted as minimizing the variance of the representative features.

In this study, a tight regularization is also proposed to further restrict the spread range of defect-free samples in the CAE model training. It is given by

$$L_d^{(l)}(X) = (1 - \lambda)\frac{1}{N}\sum_{i=1}^{N}\left\| x_i - \hat{x}_i^{(l)} \right\|^2 + \lambda\frac{1}{N}\sum_{i=1}^{N}max\left\{ d_i^{(l)} - \overline{d}^{(l-1)}, 0 \right\}^2 \tag{10}$$

where

$d_i^{(l)} = \left\| V_i^{(l)} - \overline{V}^{(l-1)} \right\|$;
$\overline{d}^{(l-1)} = \frac{1}{N}\sum_{i=1}^{N}d_i^{(l-1)}$

Table 1
Detailed parameter setting of the CAE models: (a) encoder; (b) decoder.

(a) Encoder				
Number	Layer	Number of filters	Filter size	Activation function
1	Convolution (s = 1)	64	3*3	Relu
2	Convolution (s = 2)	32	3*3	Relu
3	Convolution (s = 1)	10	3*3	Relu
4	Convolution (s = 2)	10	3*3	Relu
5	Convolution (s = 1)	10	3*3	Tanh

(b) Decoder				
Number	**Layer**	**Number of filters**	**Filter size**	**Activation function**
6	Flatten			
7	Dense	Units: 6250		Sigmoid
8	Convolution*	10	3*3	Relu
9	Conv2DTranspose	10	3*3	Relu
10	Convolution*	32	3*3	Relu
11	Convolution*	32	3*3	Relu
12	Convolution*	64	3*3	Relu
13	Convolution*	1	3*3	Sigmoid

*s = strides (Number of filters in layer 5 could be changed for feature vectors.).
*Zero-padding is applied.

$d_i^{(l)}$ is the distance of training sample x_i to its center at epoch l, and $\overline{d}^{(l-1)}$ is the mean distance to the center from epoch l-1. The training sample with a small distance $d_i^{(l)}$ below the mean $\overline{d}^{(l-1)}$ is discarded. The backpropagation updating can then focus more on the training samples with distances larger than the mean. This regularization can further reduce the upper bound of the distances and makes all training samples fall within a tight cluster in the distance domain. It can be expected that a defective sample will generate a significantly large distance from the cluster center of normal samples.

3.3. Classification of anomalies

For the extracted features from CAEs, the SPC (Statistical Process Control) thresholding is applied as a one-class classifier to identify defective samples. The proposed regularization in the loss function tends to minimize the spread (i.e. variance) of the extracted features. The simple SPC is thus a natural selection to set up the discrimination threshold.

The SPC threshold of distance d_I for a test image I is given by

$$T_d = \mu_d + C \cdot \sigma_d \tag{11}$$

where

$d_I = \left\| V_I - \overline{V}_X \right\|$;

$\mu_d = \frac{1}{N}\sum_{i=1}^{N}d_i = \frac{1}{N}\sum_{i=1}^{N}\left\| V_i - \overline{V}_X \right\|$;

$\sigma_d = \left\{ \frac{1}{N}\sum_{i=1}^{N}(d_i - \mu_d)^2 \right\}^{1/2}$;

V_i is the feature vector of normal sample x_i in the training set X;
$\overline{V}_X$ is the mean feature vector of the training set X;
C is a control limit constant

The test image I is identified as defective if $d_I > T_d$. Otherwise, it is defect-free.
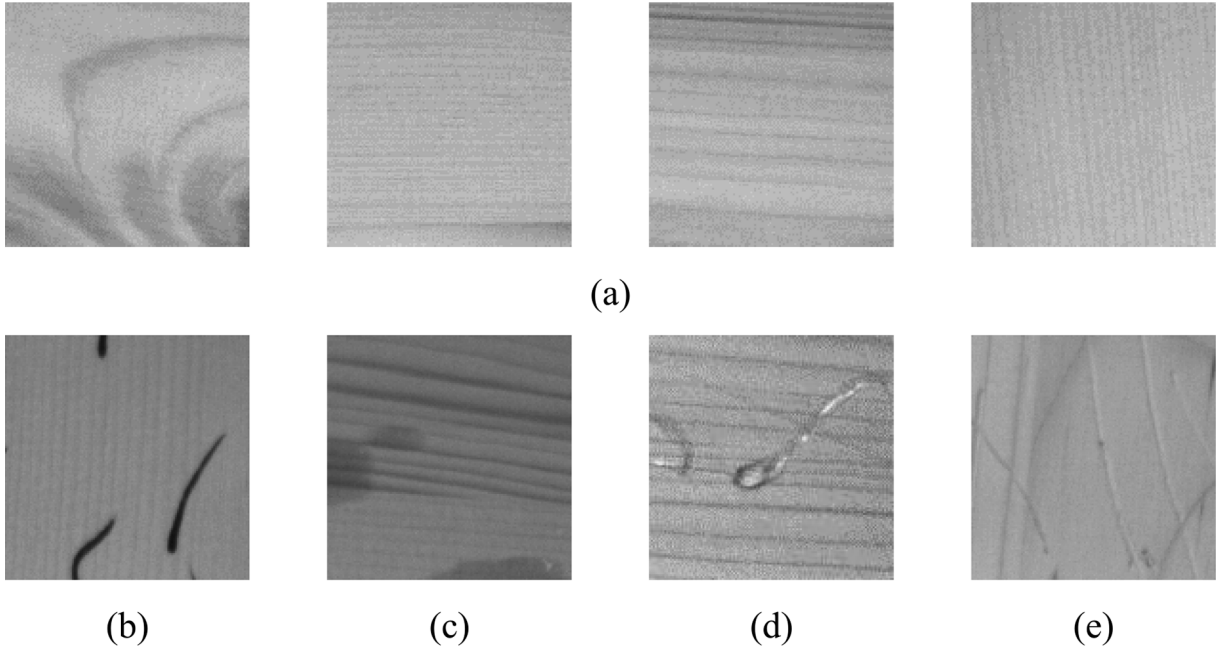
(a)

(b)   (c)   (d)   (e)

Fig. 2. Natural wood examples for the evaluation: (a) 4 defect-free surface images; (b) contaminant; (c) oil-stain; (d) glue; (e) scratch.

4. Experimental results

In this section, the performance of the CAE models with and without regularization is evaluated for defect detection in various material surfaces. The proposed models and methods are implemented with Tensorflow and Keras. They are executed on a PC equipped with Intel i7-7700 3.40 GHz CPU and one NVIDIA GTX 1080 Ti GPU. The CAE architecture used for feature extraction has been shown in Fig. 1. The detailed setting of each layer in the CAE models is presented in Table 1.

The Adam optimizer with a learning rate = 0.001, batch size = 128 and epochs = 1000 has been applied to the conventional CAE and the regularized CAE models. The test sample images used in the experiments are provided by local manufactures. They are available upon request from the authors. Public image datasets used for industrial optical inspection are also evaluated.

4.1. Effect of the regularization weight λ

The proposed loss function for λ-CAE model training involves a penalty parameter λ for regularization. Thus, the effect of changes in λ value on the defect detection results is evaluated. When $\lambda = 0$, no regularization is performed and only the reconstruction error (i.e. the traditional CAE) is taken into consideration.

The natural wood surfaces are used for the evaluation. Fig. 2(a) presents four normal wood surface images that contain irregular structure-textures. Some involve knots that could be falsely identified as defects. Fig. 2(b)–(e) are defective test images that involve contaminant, oil-stain, glue and scratch. The wood images were taken by an IDS UI-3360CP camera from a working distance of 360 mm with coaxial lighting. The image resolution is 165 μm/pixel. The input image to the CAE models is of size 128 × 128 pixels, which corresponds to 21.1 × 21.1 mm^2.

A total of 1900 defect-free images, each of size 128 × 128 pixels, are used to train the standard CAE (Eq. (6)) and the proposed λ-CAE with loss function L_V (Eq. (9)). The test images for the evaluation include 100 normal wood images and 200 defective wood images (50 samples for each defect type). Five feature maps, each of size 8 × 8, from the encoders are used to form a 320-dimensional feature vector $\boldsymbol{V}$ for the CAEs. The regularization factor λ is varied from 0, 0.05, 0.1, 0.2, 0.5 to 1. Note
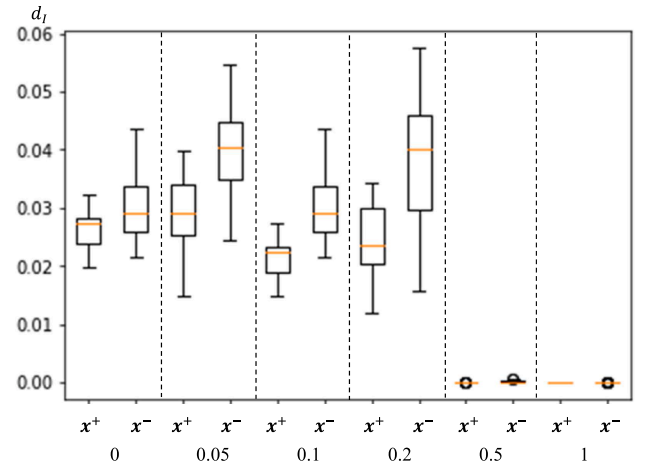

Fig. 3. Box-plots of defect-free (x^+) and defective (x^-) wood surface samples for the CAE ($\lambda=0$) and the proposed λ-CAE with loss L_V under various λ values. The value in the y-axis is the distance d_I.

that the standard CAE is applied when $\lambda = 0$. Fig. 3 presents the box-plots of the distance $d_{I_j} = \left\| V_{I_j} - \overline{V}_X \right\|$ in the y-axis for the λ-CAE model with varying λ values, where V_{I_j} is the feature vector of the test sample I_j and $\overline{V}_X$ is the mean feature vector of the training set $\boldsymbol{X}$.

The box-plot shows the 50th percentile (median), 25th percentile and 75th percentile (and minimum and maximum) of the test data. The

Table 2
Effect of various λ values in the loss L_V for wood defect detection.

λ	0 (CAE)	0.05	0.1	0.2	0.5	1
FP rate	38%	35%	22%	31%	50%	21%
	(38)	(35)	(22)	(31)	(50)	(21)
FN rate	50%	34%	4%	47%	63%	66%
	(100)	(68)	(8)	(64)	(127)	(132)
Accuracy	54%	65%	90%	68%	41%	49%

*100 defect-free and 200 defective test images.

Table 3
Effect of various λ values in the loss L_d for wood defect detection.

λ	0 (CAE)	0.05	0.1	0.2	0.5	1
FP rate	38%	33%	8%	32%	25%	36%
	(38)	(33)	(8)	(32)	(25)	(36)
FN rate	50%	30%	4%	26%	65%	66%
	(100)	(60)	(8)	(52)	(131)	(132)
Accuracy	54%	69%	95%	72%	48%	44%

*100 defect-free and 200 defective test images.

Table 4
Effect of various numbers of filters for feature extraction in defect detection: (a) standard CAE model; (b) λ-CAE model with loss L_V. (Filter size: 8×8).

(a) CAE					
No. of filters	1	2	3	5	10
FP rate	36%(36)	34%(34)	36%(36)	38%(38)	38%(38)
FN rate	64%(128)	69%(138)	60%(120)	50%(100)	60%(120)
Accuracy	45%	42%	48%	54%	47%

(b) λ-CAE-L_V					
No. of filters	1	2	3	5	10
FP rate	36%(36)	35%(35)	38%(38)	22%(22)	28%(28)
FN rate	60%(121)	60%(121)	53%(106)	4%(8)	4%(8)
Accuracy	47%	48%	52%	90%	88%

results reveal that the box-plots between defect-free and defective test samples can be best separated when $\lambda = 0.1$. It far outperforms the standard CAE (with $\lambda = 0$). When $\lambda \geqslant 0.5$, the proposed λ-CAE with loss L_V cannot distinguish the difference between defect-free and defective samples due to the blurred reconstruction. By using the SPC with control limit constant $C = 2$ to set up the threshold T_d for the distance d_{I_j}, Table 2 summarizes the False Positive (FP) rate, False Negative (FN) rate and the overall detection accuracy (i.e. [TP + TN]/[(TP + FP)+(TN + FN)]) of the 300 wood test images from the λ-CAE model with varying λ values.

It shows that the standard CAE ($\lambda=0$) performs poorly with a low detection accuracy of 54%. The proposed λ-CAE with loss L_V detects most of the true defects with a very low FN rate of 4%. The overall detection accuracy is increased to 90%. The experimental results indicate that the proposed λ-CAE model can indeed make the extracted feature vectors of defect-free training samples tightly concentrate around the mean feature vector. It causes the defective samples under test far away from the cluster center of the defect-free training samples. Table 3 further presents the effect of various λ values for the λ-CAE with loss L_d. It also indicates that $\lambda = 0.1$ yields the best detection accuracy of 95%.

4.2. Effect of the number of filters (dimension of **V**)

The number of filters used in the last convolutional layer of the encoder determines the number of the corresponding feature maps and, thus, the dimension of the feature vector **V**. Let $r \times c$ be the feature-map size, and K the number of filters used for the convolution. The size of the feature vector is then $r \cdot c \cdot K$. The same wood images described above in Sec. 4.1 are also used for the evaluation. The number of filters is varied from 1, 2, 3, 5 to 10 with λ value fixed at 0.1. Table 4(a) and (b) presents the detection statistics of the standard CAE and the proposed λ-CAE with loss L_V, respectively.

The SPC threshold with control constant $C = 2$ is used for the discrimination. The results show that the proposed λ-CAE performs far better than the standard CAE, regardless of the number of filters. For the proposed λ-CAE, insufficient number of filters (such as 1, 2 or 3 filters for low-dimensional feature vectors) generates a low accuracy rate around 50%. When the number of representative feature maps is large enough, the detection accuracy can be boosted up to 90% with 5 filters. In the
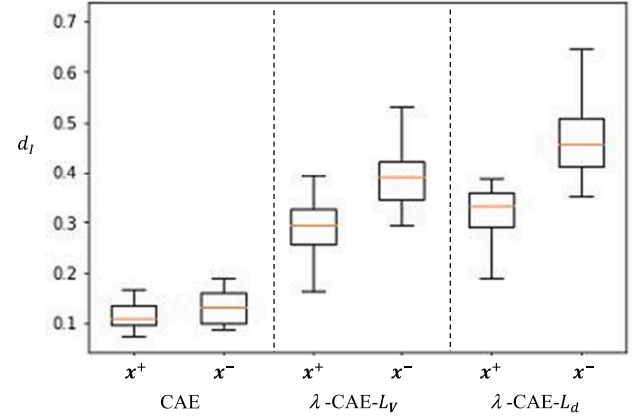

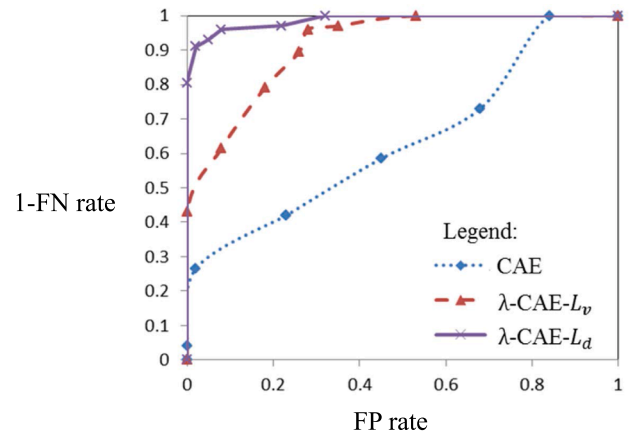**Fig. 4.** Box-plot of defect-free (x^+) and defective (x^-) wood test samples for the three CAE models, where the y-axis shows the distance $d_{I_j} = \left\| V_{I_j} - \overline{V}_X \right\|$.

Fig. 5. ROC curves of the CAE, λ-CAE with L_V and λ-CAE with L_d for the natural wood test samples.

wood test examples, 10 convolutional filters yield a detection accuracy of 88%. Excessive feature maps may not generate better detection accuracy.

4.3. Comparison of loss functions L_V and L_d

In section 3, two regularizations (loss functions L_V and L_d) have been proposed to improve the CAE model for anomaly detection. The wood test examples described in Section 4.1 are used for the evaluation. The same CAE architecture with the same hyperparameter setting is used for the standard CAE and the λ-CAE models with loss L_V and L_d. For the λ-CAE Models, the regularization weight λ is fixed to 0.1. Fig. 4 presents the box-plots of the three comparative CAE models, where the distance d is shown in the y-axis.

The results show that the λ-CAE model with loss L_d gives the best discrimination between defective and defect-free test images. The Fisher's ratios (between-class variance/within-class variance) are 0.14, 1.54 and 2.90 for the standard CAE, λ-CAE with loss L_V, and λ-CAE with loss L_d, respectively. The discrimination power of the proposed λ-CAE models is significantly higher than the conventional CAE model. Fig. 5 further illustrates the ROC (Receiver Operating Characteristic) curves of the three comparative CAE models. The AUCs (Area Under Curves) are respectively given by 0.70, 0.89 and 0.96 for the standard CAE, λ-CAE with loss L_V, and λ-CAE with loss L_d. The λ-CAE model with loss L_d generates an ROC with the AUC close to an ideal value of 1.

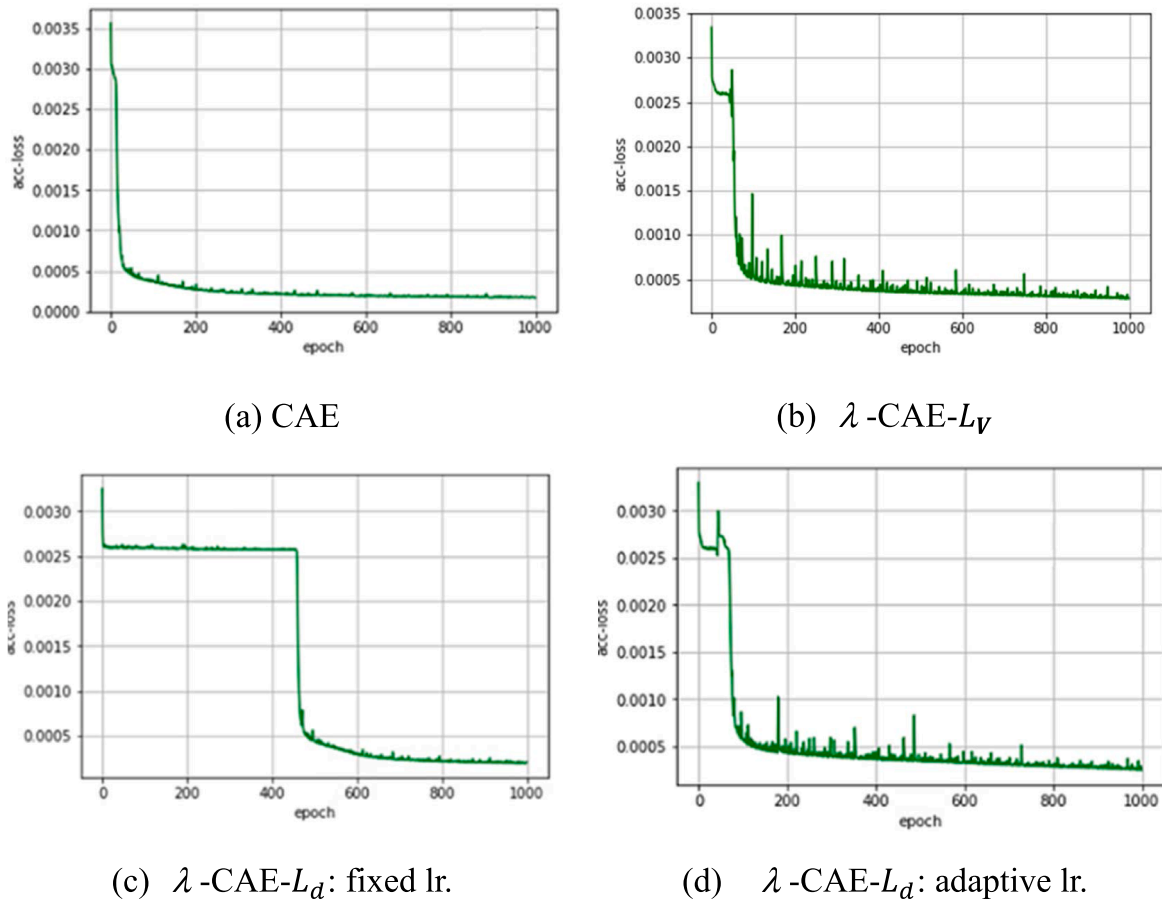In terms of optimization convergence of the CAE modeling training,

(a) CAE

(b) λ -CAE-L_V

(c) λ -CAE-L_d: fixed lr.

(d) λ -CAE-L_d: adaptive lr.

Fig. 6. Loss function over training epochs: (a) CAE; (b) λ-CAE with L_V; (c) λ-CAE with L_d trained by a fixed learning rate; (d) λ-CAE with L_d trained by adaptive learning rate.

Table 5
Performance comparison of the wood test samples for the three CAE models (based on 5 filters and $\lambda = 0.1$).

Model	FP rate	FN rate	Accuracy
CAE	38%(38)	50%(100)	54%
λ-CAE-L_V	22%(22)	4%(8)	90%
λ-CAE-L_d	8%(8)	4%(8)	95%

*100 defect-free and 200 defective test images.

Fig. 6(a)–(c) presents the loss function values over training epochs. All three models used a fixed learning rate of 0.001. The curves show that the standard CAE model is converged fast and stable. The proposed λ-CAE with loss L_V is unstable to convergence in early epochs. This could be due to the calculation of the mean feature vector $\overline{V}^{(l-1)}$ from the previous epoch. The λ-CAE model with loss L_d is even more difficult to converge. It gets stuck for a few hundreds of epochs and then is eventually stabilized and converged by using a fixed learning rate. Fig. 6(d) shows the loss function over epochs for the λ-CAE model with loss L_d by setting a large learning rate of 0.1 in the early 100 epochs and then a fixed small learning rate of 0.001 afterwards. This setup can prevent the bottleneck of convergence. A more dedicated and adaptive learning rate strategy could further improve the training for the λ-CAE model with loss L_d. The experiments also show that the learning rate generally does not affect the detection accuracy as long as it can converge in the training process by observing the loss function value over epochs.

The experimental results for the wood samples from Table 2, Table 4 and Fig. 4 are summarized in Table 5 to compare the performance of the regularized CAE and the standard CAE. The statistics are based on $\lambda =$ 0.1 and 5 filters at the last layer of the encoder. The results show that the proposed CAE with loss L_d yields a detection accuracy of 95%, which is far better than 54% given by the standard CAE.

4.4. Detection on various material surfaces

This subsection presents the defect detection results of the standard CAE and the proposed λ-CAE with loss L_V and L_d on various material surfaces. All CAE models have the same neural network architectures and the same hyperparmater settings for various material surfaces. 5 filters are used in the last convolutional layer of the encoder to extract the features for all CAE models and for all material surfaces. In the subsequent experiment, the detection of anomalies based on the extracted features from the CAE models is measured by the SPC thresholding. The control limit constant C for the SPC is set to 2 for all test samples of various material surfaces. The same optimizer is used to train individual CAE models. The material surfaces used for the test include textured surfaces (LCDs and PCB fiberglass substrates) and patterned surfaces (medical acne patches). They cover many types of material surfaces found in the manufacturing industry. For the λ-CAE models, λ is fixed to 0.1 for all material surfaces. The test samples of LCDs, PCB fiberglass and medical acne patches, along with woods above, are provided by local manufacturers. The images are available upon request from the authors.

A. Liquid crystal display (LCD) with homogeneously structured texture

Fig. 7(a) and (b) demonstrates defect-free and defective LCD surfaces, respectively. The LCD images were taken by a Sony CCD camera
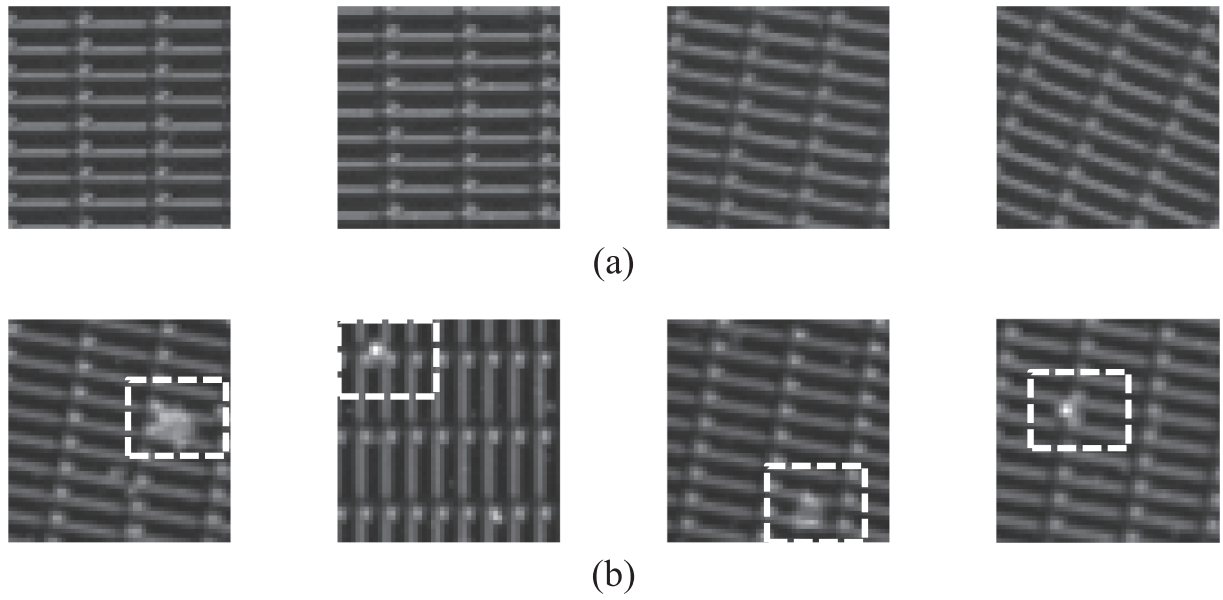
(a)

(b)

Fig. 7. Liquid Crystal Display (LCD) test examples: (a) 4 defect-free panel surfaces; (b) defective samples (marked by dot-frames).

Table 6

Detection results on the LCD test samples by CAE, λ-CAE with loss L_V, and λ-CAE with loss L_d.

Model	FP rate	FN rate	Accuracy
CAE	11%(11)	6%(2)	90%
λ-CAE-L_V	0%(0)	0%(0)	100%
λ-CAE-L_d	0%(0)	0%(0)	100%

*100 defect-free and 33 defective test images.

with two bar-shaped LED lights for the illumination. The image resolution is 16.6 μm/pixel.

The LCD panel is composed of orthogonal data lines and gate lines and results in homogeneously structured textures in the image. The defects could be particles, pinholes or contaminants in the LCD surfaces. The image patch size for the experiment is 28×28 pixels, which corresponds to $0.47 \times 0.47 \text{ mm}^2$. The feature map is 7×7. The number of defect-free LCD sample images for training the CAE models is 400. The test image patches contain 100 defect-free and 33 defective samples.

Table 6 lists the resulting FP and FN rates from the standard CAE and the proposed λ-CAE with loss L_V and L_d. For the highly regular texture-surfaces, the standard CAE yields good detection results with FP rate = 11% and FN = 6%. The proposed λ-CAE model with loss L_V or L_d improves the performance without false alarm and missing detection for all defect-free and defective LCD test samples.

B. PCB fiberglass substrates with less-regular background texture

Fig. 8(a) and (b) shows, respectively, defect-free and defective woven fiberglass substrates of printed circuit boards (PCBs). The PCB fabric weaves present also structural textures in the image. The surface patterns are relatively blurred with some structure variations, compared to the LCD patterns above. As observed closely from the demonstrative images, the test samples contain varying structural patterns. This is because the test samples were collected from different machine vision systems with different image resolutions in various production lines. The images were taken by a line-scan 8 K mono camera with line-bar lighting for illumination. The working distance was around 400 mm, depending
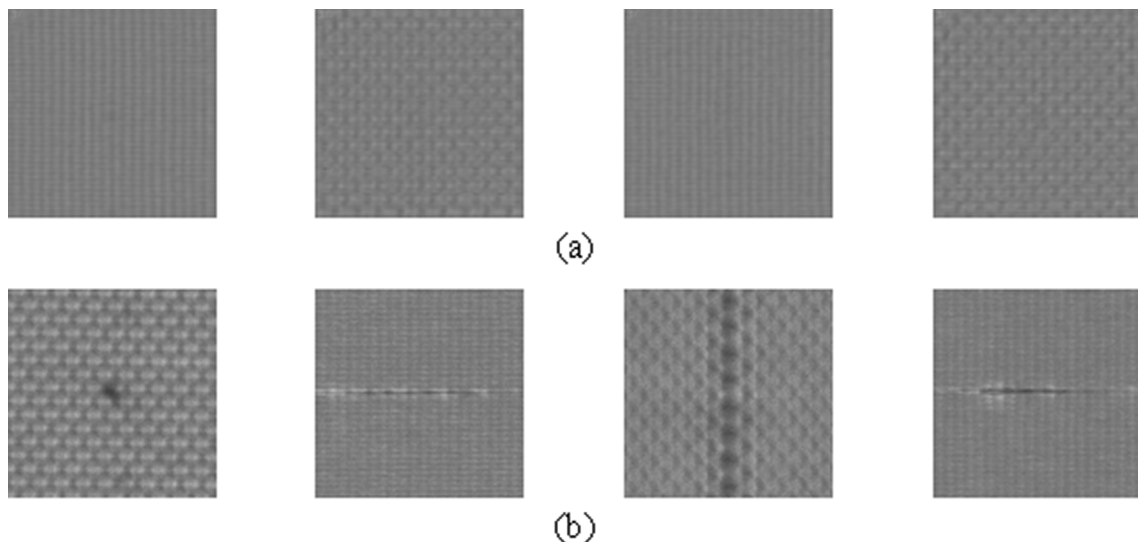

(a)

(b)

Fig. 8. PCB fiberglass substrates: (a) 4 defect-free woven surfaces; (b) defects of various types on fabric weaves.

Table 7

Detection results on the PCB fiberglass substrates by CAE, λ-CAE with loss L_V, and λ-CAE with loss L_d.

Model	FP rate	FN rate	Accuracy
CAE	7%(7)	44%(88)	68%
λ-CAE-L_V	4%(4)	25%(50)	82%
λ-CAE-L_d	4%(4)	21%(41)	85%

*100 defect-free and 200 defective test images.

on the manufacturing sites. The test image is of size 100×100 pixels, which corresponds to 8×8 mm^2. The image resolution is 80 μm/pixel.

The defects could be blob-shaped in random locations, or line-shaped along the warp or weft direction in the woven substrates. The input image size to the CAE models is 100×100 pixels. The feature map is of size 13×13 for feature extraction. A total of 500 defect-free samples are used for the model training. The test images for the evaluation contain 100 defect-free and 200 defective samples. Table 7 presents the resulting FP and FN rates from the standard CAE and the proposed λ-CAE with loss L_V and L_d. The results show that the standard CAE yields a low FP rate 7% with a poor FN rate of 44%. The proposed λ-CAE models give also a low FP rate of 4%, and improve the FN rate down to 21%. The detection accuracy is improved from 68% up to 85%.

The experiment has revealed that the proposed λ-CAE model can indeed improve the discrimination for anomaly detection, even the defect-free training samples involve a variation in the texture background. It is believed that the CAE-based anomaly detection can be further improved if the CAE models are individually trained with the texture samples from the same imaging system.

C. Medical acne patches with non-textural patterns

The last demonstrative examples are medical acne patches. Fig. 9(a) and (b) shows, respectively, defect-free and defective samples. The patterned acne patch is of circular-shaped without textures on the surface. The defective patches could involve contaminants of various sizes at random locations. Some defects are around the edges of the circular pads. The non-textured pattern images with subtle small defects are highly challenging for anomaly detection when only defect-free samples are used for the training. The IDS UI-3360CP camera with 25 mm lens at a working distance of 150 mm was used to take the image. Coaxial lighting was used for illumination. A 300×300 image corresponds to 10×10 mm^2 in physical size. The image resolution is 33 μm/pixel.
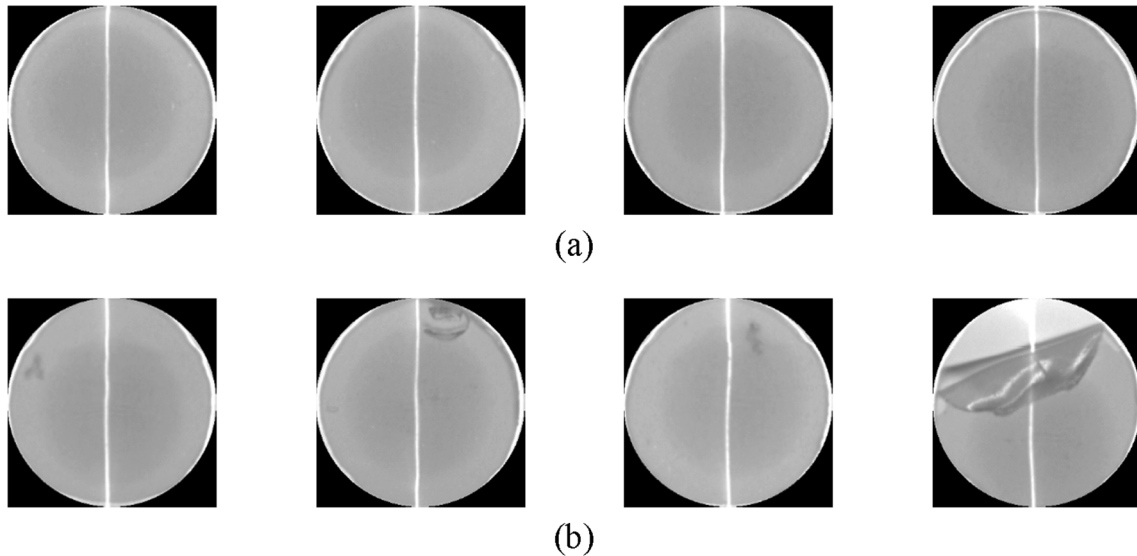
The acne patch image to the CAE models is of size 300×300 pixels. The feature-map size is 25×25 for feature extraction. A total of 400 defect-free samples are used for the model training. The test samples for the evaluation contain 100 defect-free and 500 defective images. Table 8 lists the resulting FP and FN rates from the standard CAE and the proposed λ-CAE with loss L_V and L_d. It shows that the standard CAE and the proposed λ-CAE with loss L_V yield the same FN rate of 20%. However, the λ-CAE model with loss L_V improves the FP rate from 10% to a low 1%. The λ-CAE with loss L_d further improves the detection accuracy from 81% to 86%.

D. Public DAGM datasets

The DAGM competition datasets available on Kaggle for industrial optical inspection are also evaluated in the experiment. The DAGM datasets contain textured surfaces with tiny defects in 512×512 images. Two DAGM datasets that involve structural texture and random texture are analyzed. The representative DAGM textured images for the evaluation are demonstrated in Fig. 10. The full 512×512 image size is used for the experiments. Each textured pattern is trained with 900 defect-free samples, and is evaluated with 100 defect-free and 100 defective samples. The performance of the conventional CAE and the regularized CAE models for the 2 DAGM datasets is present in Table 9. The proposed methods can detect small defects in a large textured image. The results show that the regularized CAE models can improve the detection accuracy by 5% to 10% over the conventional CAE model for the two DAGM datasets.

E. Comparison with other anomaly detection methods

The proposed methods (CAE with regularizations L_v and L_d) are compared to Deep SVDD (Support Vector Data Description) (Refs.

Table 8

Detection results on the medical acne patches by CAE, λ-CAE with loss L_V, and λ-CAE with loss L_d.

Model	FP rate	FN rate	Accuracy
CAE	10%(10)	20%(100)	81%
λ-CAE-L_V	1%(1)	20%(100)	83%
λ-CAE-L_d	1%(1)	16%(79)	86%

*100 defect-free and 500 defective test images.

(a)

(b)

Fig. 9. Patterned acne patches: (a) 4 defect-free sample images; (b) defective sample images.
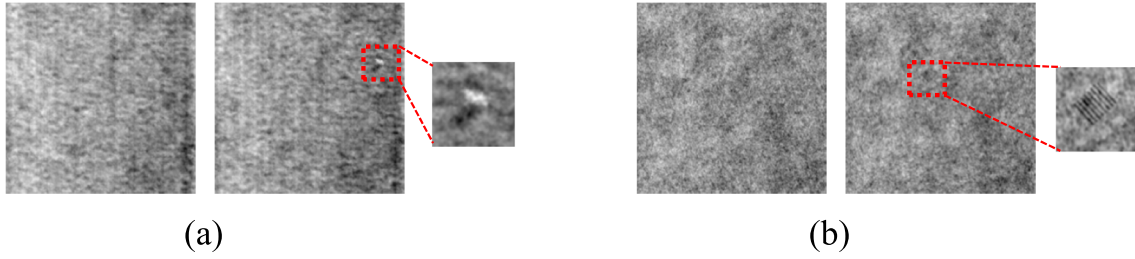
Fig. 10. DAGM datasets for the evaluation: (a) structural textures (defect-free vs. defective); (b) random textures (defect-free vs. defective).

Table 9
Detection results on the DAGM datasets by CAE, λ-CAE with loss L_V, and λ-CAE with loss L_d.

(a) Structural-texture background (Fig. 10(a))

Model	FP rate	FN rate	Accuracy
CAE	23%(23)	16%(16)	80%
λ-CAE-L_V	16%(16)	7%(7)	88%
λ-CAE-L_d	12%(12)	7%(7)	90%

(b) Random-texture background (Fig. 10(b))

Model	FP rate	FN rate	Accuracy
CAE	12%(12)	8%(8)	90%
λ-CAE-L_V	7%(7)	6%(6)	94%
λ-CAE-L_d	7%(7)	4%(4)	95%

*100 defect-free and 100 defective test images.

Table 10
Performance comparison of anomaly detection methods on various test datasets. (CAE models are based on 5 filters and $\lambda = 0.1$).

(a) Natural woods (Fig. 2)

Model	FP rate	FN rate	Accuracy
Deep-SVDD	30%(30)	55%(111)	53%
CAE	38%(38)	50%(100)	54%
λ-CAE-L_V	22%(22)	4%(8)	90%
λ-CAE-L_d	8%(8)	4%(8)	95%

(b) Medical acne patches (Fig. 9)

Model	FP rate	FN rate	Accuracy
Deep-SVDD	8%(8)	32%(162)	72%
CAE	10%(10)	20%(100)	81%
λ-CAE-L_V	1%(1)	20%(100)	83%
λ-CAE-L_d	1%(1)	16%(79)	86%

(c) DAGM structural-texture background (Fig. 10(a))

Model	FP rate	FN rate	Accuracy
Deep-SVDD	78%(78)	26%(26)	48%
CAE	23%(23)	16%(16)	80%
λ-CAE-L_V	16%(16)	7%(7)	88%
λ-CAE-L_d	12%(12)	7%(0)	90%

(d) DAGM random-texture background (Fig. 10(b))

Model	FP rate	FN rate	Accuracy
Deep-SVDD	76%(76)	18%(18)	53%
CAE	12%(12)	8%(8)	90%
λ-CAE-L_V	7%(7)	6%(6)	94%
λ-CAE-L_d	7%(7)	4%(4)	95%

(e) MVTech carpet images (Fig. 11)

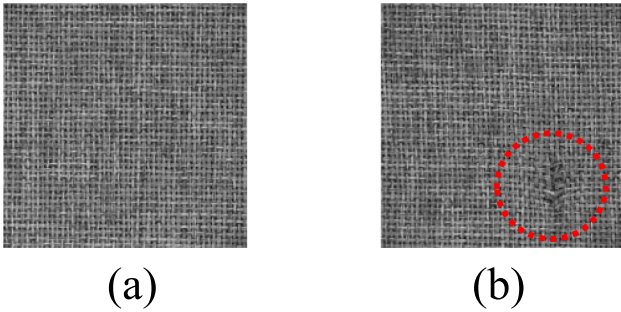| Model	FP rate	FN rate	Accuracy
Deep-SVDD	14%(4)	20%(18)	81%
CAE	18%(5)	21%(19)	80%
λ-CAE-L_V	7%(2)	18%(16)	85%
λ-CAE-L_d	0%(0)	12%(11)	91%

Fig. 11. MVTech carpet dataset for the evaluation: (a) defect-free image; (b) cut- defect sample.

[29,30]). The objective of Deep-SVDD is to learn a convolutional neural network that minimizes the volume of a data-enclosing hypersphere from normal samples. Points mapped outside the sphere are identified as anomalies. It has been reported that Deep-SVDD outperforms (or is competitive with) AnoGAN (Refs. [37,38]) and DCAE (Ref. [32]) on MNIST (handwritten digits), CIFAR-10 (colored objects), and CT-SRB (stop signs) image datasets. The comparison of different anomaly detection methods is based on natural woods (Fig. 2), patterned acne patches (Fig. 9), DAGM datasets (Fig. 10), and MVTech AD carpet dataset (Fig. 11). For the public MVTech carpet dataset, the image size is 128×128 pixels. A total of 280 defect-free carpet images are used for the training. The test samples involve 28 normal images and 89 defective carpet images containing color, cut, hole, contamination and thread defects. The resulting statistics in Table 10 show that Deep-SVDD does not perform well on the surfaces with random and irregular textures. The detection accuracy is only around 50% for the DAGM images. It performs better on the patterned acne patch images with an accuracy rate of 72% and the MVTech carpet images with an accuracy rate of 81%. The proposed CAE with loss L_V or L_d outperforms Deep-SVDD for all textured and patterned surfaces in defect detection. The comparison

of CAE/VAE with pixel-wise reconstruction (Refs. [33,34,36]) and CAE with feature extraction for unsupervised defect detection is also analyzed in Appendices A and B.

F. Qualitative analysis of detection results on various textured surfaces

Based on the experimental results of various textured surfaces above, the qualitative analysis is discussed as follows. For defect detection in a highly-regular surfaces (e.g. LCDs) that shows no variation in the texture background, the proposed λ-CAE with loss L_V can generate feature vectors spread in a very tight region and results in reliable and accurate detection. For the homogeneously textured surfaces (e.g. wood images and DAGM datasets), the proposed λ-CAE with loss L_d yields similar representative features distributed in a limited range for defect-free training samples and yields very good detection accuracy. For the PCB fiberglass surfaces, they present various structural patterns with
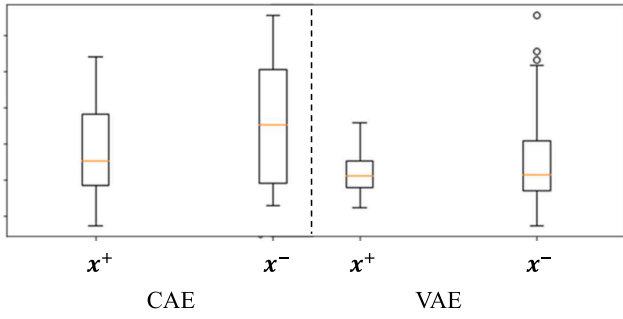
Fig. A1. Box-plots of defect-free (x^+) and defective (x^-) wood surface samples for the CAE and VAE models.
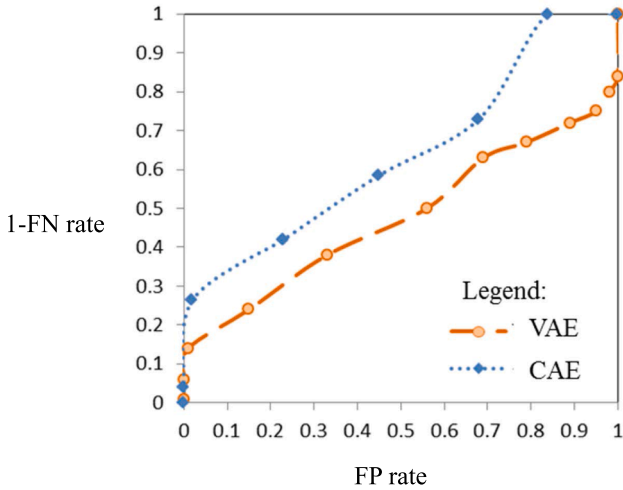

Fig. A2. ROC curves of CAE and VAE models for the natural wood test samples.

different image resolutions. The training samples thus cause a relatively larger variation of the extracted features from the λ-CAE models and result in a larger FN rate of 20%. For on-site implementation, the training samples to the λ-CAE models should contain the same texture background at a fixed image resolution to minimize the variation of extracted features.

For the patterned acne patches, they do not show repetitively textured pattern everywhere in the image. This may cause the anomaly detection for small defects in the patterned images very difficult. The proposed λ-CAE models can generate a low FP rate less than 1%. But, the FN rate can be as high as 16% since some defects are very small w.r.t. the 300×300 input image or around the circular edges. To detect small defects in a patterned surface, the original source image could be divided into small subimages (e.g. divide the image into quarters). Each subimage region is trained with an individual λ-CAE model. This could reduce the spread of representative features from the defect-free training samples and make a local small defect relatively significant in size w.r.t. the divided subimage region.

5. Conclusions

The CAE has been a very popular unsupervised model to extract features for anomaly detection. However, the extracted features from the traditional CAE could be widely spread and degrade the performance for unsupervised defect detection. The main contribution of the paper is the two new regularization penalties that can effectively confine the spread of the extracted features in a very limited space from a set of defect-free samples. The first regularization is very effective and easy to train for homogeneously-textured surfaces. The second regularization can further discriminate the defect features for less-regular texture backgrounds. Both regularizations far outperform the traditional CAE for textured surfaces with structural or random patterns. They also work well for patterned surfaces that contain small, subtle defects. Besides the new regularizations proposed in the CAE loss function, the superiority of CAE over VAE in feature extraction and the performance of CAE with feature extraction and image reconstruction for unsupervised defect detection applications are also analyzed and reported in the appendices.

Compared to the conventional machine learning methods that require hand-crafted features, the proposed autoencoder models can automatically extract the representative features for the inspection image with textured surfaces or non-periodic patterns. The proposed regularizations to the CAE models can improve the performance of the conventional CAE model in both false positive and false negative rates. The loss L_d with regularization $\sum max\{d_i - \bar{d}, 0\}^2$ outperforms the loss L_V with regularization $\sum \left\| V_i - \overline{V} \right\|^2$ in detection accuracy, but it is relatively less stable to train. Adaptive learning with a large learning rate in early epochs and a small learning rate in late epochs can stabilize the convergence.

The test results on a variety of material surfaces, from highly regular-texture surfaces, random/structured texture surfaces to complicated pattern surfaces, have shown that the proposed λ-CAE models with regularizations can effectively identify the unseen defects. The proposed models can improve the detection accuracy by 5–17%, compared to the conventional CAE model. The CAE-based models are computationally very efficient in the inspection process. They require only 3 ms to process a 512×512 image on a typical PC with one low-end GPU.

The proposed methods can be beneficial to industrial inspection applications, where the defect samples are rare or not available in the early manufacturing stage of new products. Although the proposed λ-CAE models mainly aim at defect detection in the manufacturing industry, they can potentially be extended for anomaly detection in various fields, where only normal samples are available for the model training. The proposed method is image-wise defect detection, which classifies a test image as defective or defect-free. However, it can be extended to detect the location of a local defect in the textured image by sliding a small window pixel-by-pixel (or with a small stride) over the entire test image. The windowed patch image is then the input to the CAE models. The central pixel of the patch is then marked as defective or defect-free based on the CAE detection result.

The regularizations proposed in the paper require the information of the mean feature vector $\overline{V}$ at each training epoch. It is currently realized from all training samples in the previous epoch. That is, the $\overline{V}$ updating is deferred by one epoch. The loss function during training is thus more fluctuated, compared to the original CAE model that evaluates only reconstruction errors. It is worth further investigation for the evaluation of mean feature vector from the current epoch. For example, the limitation of the currently proposed regularizations on the mean feature vector $\overline{V}$ could be released by treating $\overline{V}$ as an unknown variable vector. The convergence of the model training could be more difficult than the currently proposed model that calculates $\overline{V}$ from the previous epoch. Dynamic and automatic learning rates for efficient and effective training of λ-CAE models are also demanded in the future.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.
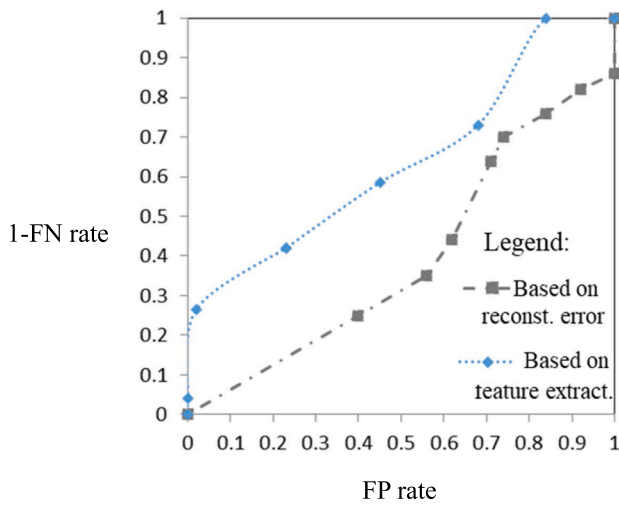
Fig. B1. ROC curves of CAE based on feature extraction (d_x) and reconstruction error ($\Delta\varepsilon_x$).

Appendix A. Comparison of CAE and VAE

In this appendix, the performance of the CAE and VAE models for defect detection is evaluated. A VAE model also inherits the CAE architecture, but learns the joint distribution over the input image and a set of latent variables by Bayesian inference, where the latent variables are restricted to be a standard multivariate Gaussian distribution. The wood image examples described in Section 4.1 are used for the evaluation. The training and test samples are the same as those described in the text. The feature vector is 320-dimensional for the CAE and, thus, the latent variables are set to 320 for the VAE for a fair comparison. Fig. A1 shows the box-plots of defect-free and defective test samples for the CAE and VAE. The Fisher's ratios are 0.14 and 0.09 for CAE and VAE, respectively.

Fig. A2 further presents the ROC curves of the CAE and VAE models. The ROC curve of CAE completely dominates that of VAE for defect detection in natural wood surfaces. All evaluation measures indicate the representative features extracted from the CAE perform better than the latent variables extracted from the VAE for defect detection applications when only defect-free samples are used for the model training.

Appendix B. Comparison of feature extraction and reconstruction error

This appendix evaluates the performance of feature extraction and reconstruction error for defect detection with CAE. The discriminant measure is given by the distance $d_{x_I} = \left\| V_{x_I} - \overline{V} \right\|$ for feature extraction, and $\Delta\varepsilon_{x_I} = \left\| x_I - \widehat{x}_I \right\|$ for reconstruction error, where x_I is the input test image, $\widehat{x}_I$ the reconstructed image, and V_{x_I} the extracted features from the trained CAE. The natural wood examples described in Section 4.1 are used for the evaluation. The SPC thresholdings are applied to d_{x_I} and $\Delta\varepsilon_{x_I}$ for the discrimination. Fig. B1 illustrates the ROC curves based on the feature extraction and reconstruction error. The results indicate that the ROC curve of the feature-extraction measure completely dominates that of the reconstruction-error measure. For defect detection applications that involve small defect sizes, the feature-extraction measure outperforms the reconstruction-error measure.

References

[1] X. Xie, A review of recent advances in surface defect detection using texture analysis techniques, Electron. Lett. Comput. Vision Image Anal. 7 (2008) 1–22.

[2] K. Hanbay, M.F. Talu, O.F. Ozguven, Fabric defect detection systems and methods - a systematic literature review, Optik 127 (2016) 11960–11973.

[3] S.H. Hanzaei, A. Afshar, F. Barazandeh, Automatic detection and classification of the ceramic tiles' surface defects, Pattern Recogn. 66 (2017) 174–189.

[4] L. Liu, P. Fieguth, Y. Guo, X. Wang, M. Pietikanen, Local binary features for texture classification: taxonomy and experimental study, Pattern Recogn. 62 (2017) 135–160.

[5] S.A.H. Ravanidi, N. Pan, The influence of gray-level co-occurrence matrix variables on the textural features of wrinkled fabric surfaces, J. Text. Inst. 102 (2011).

[6] F. Bianconi, A. Fernandez, Rotation invariant co-occurrence features based on digital circles and discrete Fourier transform, Pattern Recogn. Lett. 48 (2014) 34–41.

[7] G.-H. Hu, Q.-H. Wang, G.-H. Zhang, Unsupervised defect detection in textiles based on Fourier analysis and wavelet shrinkage, Appl. Opt. 54 (2015) 2963–2980.

[8] A.S. Malek, L. Bique, Optimization of automated online fabric inspection by fast Fourier transform (FFT) and cross-correlation, Text. Res. J. 83 (2013) 256–268.

[9] J. Jing, H. Zhang, P. Li, Improved Gabor filters for textile defect detection, Procedia Eng. 15 (2011) 5010–5014.

[10] F. Riaz, A. Hassan, S. Rehman, U. Qamar, Texture classification using rotation- and scale-invariant Gabor texture, IEEE Signal Process Lett. 20 (2013) 607–610.

[11] J.B. Florindo, O.M. Bruno, Texture analysis by fractal descriptors over the wavelet domain using a best basis decomposition, Physica A 444 (2016) 415–427.

[12] W.-C. Li, D.-M. Tsai, Wavelet-based defect detection in solar wafer images with inhomogeneous texture, Pattern Recogn. 45 (2012) 742–756.

[13] M. Ahmed, A.N. Mahmood, J. Hu, A survey of network anomaly detection techniques, J. Network Comput. Appl. 60 (2016) 19–31.

[14] R. Domingues, M. Fillippone, P. Michiardi, J. Zouaoui, A comparative evaluation of outlier detection algorithms: experiments and analysis, Pattern Recogn. 74 (2018) 406–421.

[15] R. Mohammadi-Ghazi, Y.M. Marzouk, O. Buyukozturk, Conditional classifiers and boosted conditional Gaussian mixture model for novelty detection, Pattern Recogn. 81 (2018) 601–614.

[16] M.S. Sadooghi, S.E. Khadem, Improving one class support vector machine novelty detection using nonlinear features, Pattern Recogn. 83 (2018) 14–33.

[17] D. Chakraborth, V. Narayanan, A. Ghosh, Integration of deep feature extraction and ensemble learning for outlier detection, Pattern Recogn. 89 (2019) 161–171.

[18] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444.

[19] J. Wang, Y. Ma, L. Zhang, R.X. Gao, D. Wu, Deep learning for smart manufacturing: methods and applications, J. Manuf. Syst. 48 (2018) 144–156.

[20] A. Krizhevsky, L. Sutskever, G. Hinton, ImageNet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (NIPS) (2012).

[21] D. Weimer, B. Scholz-Reiter, M. Shpitalni, Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection, CIRP Ann. – Manuf. Technol. 65 (2016) 417–420.

[22] F.-C. Chen, M.R. Jahanshahi, NB-CNN: deep learning-based crack detection using convolutional neural network and naïve Bayes data fusion, IEEE Trans. Ind. Electron. 65 (2018) 4392–4400.

[23] D. Soukup, R. Huber-Mork, Convolutional neural networks for steel surface defect detection form photometric stereo image, in: Int'l. Symp. on Visual Computing, 2014, pp. 668–677.

[24] H. Yang, S. Mei, K. Song, B. Tao, Z. Yin, Transfer-learning-based online mura defect classification, IEEE Trans. Semicond. Mfg. 31 (2018) 116–123.

[25] W. Dai, A. Mujeeb, M. Erdt, A. Sourin, Soldering defect detection in automatic optical inspection, Adv. Eng. Inf. 43 (2020) 101004.

[26] Y. Wang, M. Liu, P. Zheng, H. Yaung, J. Zou, A smart surface inspection system using faster R-CNN in cloud-edge computing environment, Adv. Eng. Inf. 43 (2020) 101037.

[27] S. Niu, B. Li, X. Wang, H. Lin, Defect image sample generation with GAN for improving defect recognition, IEEE Trans. Auto. Sci. Eng. 17 (2020) 1611–1622.

[28] D. Kwon, H. Kim, J. Kim, S.C. Suh, I. Kim, J.J. Kim, A survey of deep learning-based network anomaly detection, Cluster Comput. 22 (2019) 949–961.

[29] L. Ruff, R. Vandermeulen, N. Gornitz, L. Deecke, S.A. Siddiqui, A. Binder, E. Muller, M. Kloft, Deep one-class classification, in: Proc. of Intl. Conf. on Machine Learning Research, vol. 80, 2018, pp. 4393–4402.

[30] L. Ruff, R. A. Vandermeulen, N. Gornitz, A. Binder, E. Muller, M. Kloft, Deep support vector data description for unsupervised and semi-supervised anomaly detection, in: ICML Workshop on Uncertainty and Robustness in Deep Learning, 2019.

[31] M. Tschannen, O. Bachem, M. Lucic, Recent advances in autoencoder-based representation learning, in: Third Workshop on Bayesian Deep Learning (NeurIPS), 2018.

[32] A. Makhzani, B.J. Frey, Winner-take-all autoencoders, NIPS (2015) 2791–2799.

[33] J.K. Chow, Z. Su, J. Wu, P.S. Tan, X. Mao, Y.H. Wang, Anomaly detection of defects on concrete structures with the convolutional autoencoder, Adv. Eng. Inf. 45 (2020) 101105.

[34] J. An, S. Cho, Variational autoencoder based anomaly detection using reconstruction probability, Technical report, SNU Data Mining Center, 2015.

[35] C. Zhou, R.C. Paffenroth, Anomaly detection with robust deep autoencoders, in: ACM SIGKDD Conf. on Knowledge Discovery and Data Mining, 2017.

[36] H. Yang, Y. Chen, K. Song, Z. Yin, Multiscale feature-clustering-based fully convolutional autoencoder for fast accurate visual inspection of texture surface defect, IEEE Trans. Auto. Sci. Eng. 16 (2019) 1450–1467.

[37] T. Schlegl, P. Seebock, S.M. Waldstein, U. Schmidt-Erfurth, G. Langs, Unsupervised anomaly detection with generative adversarial networks to guide marker discovery, in: Int'l. Conf. on Information Processing in Medical Imaging, 2017, pp. 146–157.

[38] T. Schlegl, P. Seebock, S.M. Waldstein, G. Langs, U.M. Schmidt-Erfurth, f-AnoGAN: fast unsupervised anomaly detection with generative adversarial networks, Med. Image Anal. 54 (2019) 30–44.

[39] H. Zenati, C.-S. Foo, B. Lecouat, G. Manek, V.R. Chandrasekhar, Efficient GAN-based anomaly detection, in: Intl. Conf. on Learning Representations (ICLR), 2018.

[40] L. Deecke, R. Vandermeulen, L. Ruff, S. Mandt, M. Kloft, Image anomaly detection with generative adversarial networks, in: Joint European Conf. on Machine Learning and Knowledge Discovery in Databases, 2018.

[41] M. Arjovsky, L. Bottou, Towards principled methods for training generative adversarial networks, in: Intl. Conf. on Learning Representations (ICLR), 2017.

[42] K. Choi, M. Wu, N. Goodman, S. Ermon, Meta-amortized variational inference and learning, arXiv preprint arXiv:1902.0195v1, 2019.