

Prahar M. Bhatt¹

Center for Advanced Manufacturing,
University of Southern California,
Los Angeles, CA 90089
e-mail: praharbh@usc.edu

Rishi K. Malhan¹

Center for Advanced Manufacturing,
University of Southern California,
Los Angeles, CA 90089
e-mail: rmalhan@usc.edu

Pradeep Rajendran¹

Center for Advanced Manufacturing,
University of Southern California,
Los Angeles, CA 90089
e-mail: pradeep@usc.edu

Brual C. Shah¹

Center for Advanced Manufacturing,
University of Southern California,
Los Angeles, CA 90089
e-mail: brualsha@usc.edu

Shantanu Thakar¹

Center for Advanced Manufacturing,
University of Southern California,
Los Angeles, CA 90089
e-mail: sthakar@usc.edu

Yeo Jung Yoon¹

Center for Advanced Manufacturing,
University of Southern California,
Los Angeles, CA 90089
e-mail: yeojungy@usc.edu

Satyandra K. Gupta²

Center for Advanced Manufacturing,
University of Southern California,
Los Angeles, CA 90089
e-mail: guptask@usc.edu

Image-Based Surface Defect Detection Using Deep Learning: A Review

Automatically detecting surface defects from images is an essential capability in manufacturing applications. Traditional image processing techniques are useful in solving a specific class of problems. However, these techniques do not handle noise, variations in lighting conditions, and backgrounds with complex textures. In recent times, deep learning has been widely explored for use in automation of defect detection. This survey article presents three different ways of classifying various efforts in literature for surface defect detection using deep learning techniques. These three ways are based on defect detection context, learning techniques, and defect localization and classification method respectively. This article also identifies future research directions based on the trends in the deep learning area. [DOI: 10.1115/1.4049535]

Keywords: deep learning, inspection, defect detection, image processing, machine learning, computer-aided manufacturing, automation, artificial intelligence

1 Introduction

Detecting defects is a critical capability in manufacturing applications. Ensuring that a manufacturing process is under control and working as expected requires defect detection. Based on the nature and extent of the defects, appropriate corrective actions can be performed to ensure that process performance remains satisfactory. These actions range from replacing a tool on the machine to performing maintenance on other parts of the machine. Defect detection can be viewed as a precursor to the diagnostics phase of machine maintenance. Defect detection is also a critical part of the inspection process to accept or reject a part produced by a process or delivered by a supplier. Moreover, it can also enable part rework and repair and hence reduce material wastage. Some manufacturing processes have a feedback control system that can be used to prevent the defect formation if defects can be detected early. Defect detection is also critical for building process models

that can be used for process optimization. Historically, defect detection was performed by human experts with experience with the process. The desire to enable a higher level of automation in manufacturing operations requires automated defect detection.

Processing and analysis of the images of the surfaces with defects is one of the popular ways for detecting defects. There have been several works on automated surface defect detection using traditional image processing as well as machine learning techniques. Traditional image processing techniques can provide expected results in cases where the defect patterns on the surfaces are consistent, and the background is different from the defect. Techniques like edge-detection, thresholding in grayscale images, and image segmentation are typically used to assist defect detection in such cases. There are several works that use specialized techniques for surface defect detection [1–7]. For example, a blob detection algorithm that is used for tile surface defect detection is presented in Ref. [5]. The defect detection of a textured surface by using the feature-based histogram technique presented in Ref. [6] shows the segmentation procedure in Fig. 1.

The model-based techniques work well for images with little to no variation in terms of the defects they detect. Since in industrial settings, there are many types of uncertainties in terms of the intensity of defects to their shapes and sizes, it is necessary to develop methods that adapt to such wide variations. Learning-based methods provide

¹All authors have contributed equally to this work.

²Corresponding author.

Contributed by the Computers and Information Division of ASME for publication in the JOURNAL OF COMPUTING AND INFORMATION SCIENCE IN ENGINEERING. Manuscript received July 4, 2020; final manuscript received December 16, 2020; published online February 9, 2021. Special Editor: Yan Wang.

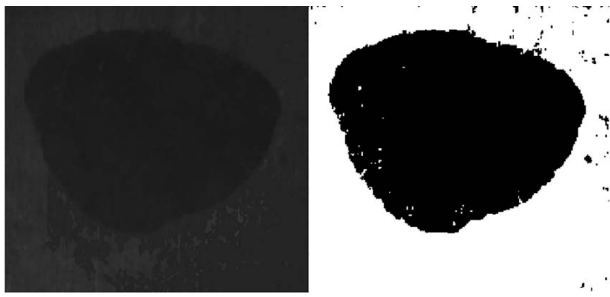


Fig. 1 The segmentation using histogram-based texture features of a fray defect on magnetic tile defect database (data [8])

a better alternative to preprogrammed feature detection methods because of the robustness to variation they provide. Classical machine learning methods for classification and regression can provide such robustness. These learning-based methods use support vector machines (SVMs) [9,10], K-nearest neighbors and Naive Bayes [11], neural networks [12], and decision trees [13]. These methods take into account statistical variations of the defects in the images to learn the desired defects. One of the major drawbacks of such methods is that precise models need to be developed to learn patterns in defects, and they may still not be robust enough to variations in textures, lighting, the complexity of defects, etc.

In the recent times, deep learning has proved to be exceptionally successful in object detection and classification, facial detection, pattern recognition, fault diagnosis, target tracking, and a wide variety of other image-based applications. It has proved to be robust to background, lighting, color, shape, sizes, and intensity in the detection of patterns in images. This is especially desirable when detecting complex surface defects in industrial settings. Challenges for defect detection in such wide ranges of settings have been shown in Fig. 2. Moreover, defects not only have to be detected but also there is a need to obtain the exact size and the type of defects.

Deep learning-based defect detection provides flexibility in terms of the network to detect custom defects based on the data set. Moreover, the parameters of the network learned for one network can be used for similar networks to generate high rates of success for surface defect detection. Furthermore, there is no need for a custom code needed for training different types of defects. The labeled data for different defects with the appropriate network provides a significantly flexible defect detection mechanism as described in several works discussed in this article.

A large number of articles have been published in the recent past focusing on deep learning in defect detection. This survey article

aims to provide readers a framework to categorize different methods and help them identify previously published works that are related to their needs. Defect detection can be performed using a wide variety of sensor data. To keep the scope tractable, this article will focus on image-based defect detection using deep learning. There are several survey articles published on defect detection using traditional feature detection and learning methods [15,16]. We will not cover these methods in this article. Survey articles have also published on anomaly detection using deep learning [17]. Our focus is on surface defect detection and, therefore, will need to focus on methods that are capable of classifying and locating defects in an image. There have been highly specialized application domain-based survey papers on defect detection that include pavement defects [18], flat steel surface defects [19], fabric defects [20], metal defect detection [21], industrial applications [22], and corrosion detection [23]. We are interested in exploring a wide variety of manufacturing applications in the surface detection area and make general observations of the methods used in these applications. Therefore, the focus of this survey article is different from what has been published until now. We mainly focus on applications related to inspection, quality control, and process modeling in manufacturing.

For image-based defect detection using deep learning methods, there can be several ways in which the existing literature can be classified. We have discussed three specific classifications in this article. The first classification is based on the context. Defect detection scope can widely vary based on the application contexts. In some contexts, just detecting the presence of a defect is adequate. In a different context, we may need to detect, classify, and label the defects. We have discussed this classification in Sec. 3. The second classification considered in this article is based on the type of learning method. The majority of the articles in the literature use supervised learning methods; however, unsupervised or semi-supervised methods are also being used. We have discussed this in Sec. 4. The third classification is based on architectures used to localize and classify defects. This has been discussed in Sec. 5. In Sec. 7, we discuss the important ideas that need to be considered when using deep learning for image-based surface defect detection. In Sec. 8, we discuss the conclusion to this survey.

2 Terminology

The following definitions are used in the following sections.

Artificial neural network (ANN): A computing system inspired by the biological neural network of the brains [24]. It consists

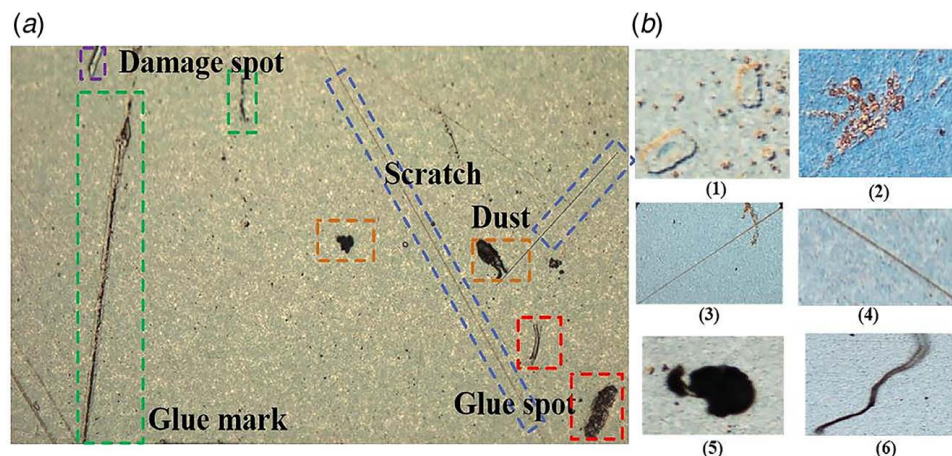


Fig. 2 The captured images of the metallic surface show challenges in defect detection: (a) defects with various shapes and sizes, (1) defects with ambiguous edges and low contrast, (2) defects with different background color, (3) fiber, (4) dust, and (5, 6) scratches [14]

of multiple layers of highly interconnected neurons (processing elements).

Shallow neural network: Neural networks have an input layer, hidden layers, and an output layer. Shallow neural networks have only one hidden layer [24].

Deep neural network (DNN): Neural networks with two or more hidden layers are called deep neural networks [24].

Autoencoders (AEs): Autoencoders [24] are a type of unsupervised neural network to learn effective data coding. It can consist of an encoder, coder, and decoder.

Convolutional neural network (CNN): CNN [24] is a type of neural network that includes a mathematical operation called convolution in one of its layers.

Generative adversarial network (GAN): GAN [25] is a machine learning model that contains two neural networks, generator, and discriminator.

Self-organizing map: Self-organizing map or self-organizing feature map [26] is a type of ANN trained using unsupervised learning. Unlike other ANNs, SOMs do not learn by back-propagation, but it uses competitive learning to adjust weights.

Softmax layer: Softmax layer [24] is a squashing function that limits the output values in the range of 0 to 1 and can be considered as a probability. The softmax layer assigns decimal probabilities to each class in multi-class classification. The total sum of the decimal values for each class equals to 1. The size of the softmax layer is the same as the output layer.

ResNet CNN: Residual networks (or ResNet) [27] is a type of ANN that speeds up the learning process by minimizing the impact of vanishing gradients. This is done by skipping the connection between all layers.

SqueezeNet CNN: SqueezeNet [28] is a smaller CNN architecture that has the same accuracy as AlexNet with 50 times fewer parameters and significantly small model size. This architecture is more suited for application requiring (a) low communication overhead for distributed training, (b) less bandwidth for exporting a new model from the cloud to the platform, and (c) deployment on limited memory hardware like Field-Programmable Gate Arrays (FPGAs).

Fully convolutional network (FCN): A neural network model that can be used for semantic segmentation. All layers are convolutional layers, and the number of channels is equal to the number of classes [29].

Fusion feature CNN (FFCNN): A CNN model consists of a feature extraction module, feature-fusion module, and decision-making module [30].

Faster region-CNN (Faster-RCNN): An improved version of RCNN by merging independent models and fastening computations [30].

DAGM data set: A data set for textured surface detection [31]

NEU database: A data set for surface defects consists of six types of defects. A database is from Northeastern University (NEU) [32]

German pattern recognition association (GAPR) data set: GAPR texture defect data set [30].

COCO data set: Common objects in context (COCO) large-scale object detection, segmentation, and captioning data set [33].

AigleRN: A training database is consisting of textured grayscale images. The images have been collected on a French pavement. The images are more complex in texture [34].

Long short-term memory (LSTM): LSTM [35] networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. The networks have an internal state that keeps past information and uses it to make predictions.

MobileNet-SSD: Mobilenet [36] is a neural network that is used for classification and recognition, whereas the SSD is a framework that is used to realize the multi-box detector. Only the combination of both can do object detection.

Thus, MobileNet can be interchanged with ResNet, inception, and so on.

Visual geometry group (VGG): VGG [37] is an innovative object-recognition model that supports up to 19 layers. Built as a deep CNN, VGG also outperforms baselines on many tasks and data sets outside of ImageNet. VGG is now still one of the most used image recognition architectures.

Region of interest (ROI): ROI [38] is a term in image processing that refers to a set of pixels in an image that is to be used for a certain image processing operation.

Meta-learning: Meta-learning [39] is a technique of accelerating learning methods by utilizing the metadata collected during a series of learning experiments.

Principle component analysis (PCA): PCA [40] is a method of data decomposition that breaks the data into a sequence of progressively less important signals that, when summed together, forms the original data.

3 Classification Based on Application Requirements and Context

This section classifies the different defect detection problems based on application requirements and context. It helps us to understand the different types of defect detection problems. We divide this section into four sections, (1) anomaly detection, (2) targeted defect detection, (3) concurrent identification of multiple defects, and (4) defect type clustering.

3.1 Anomaly Detection. Anomaly detection in defect detection is a method or a process to identify anomalies that stand for defects in data sets. It is often approached as an unsupervised learning application [41]. Anomalies are unexpected events that show deviations from normal data. Recently, researchers are increasingly using deep learning for anomaly detection. Deep learning-based anomaly detection can be categorized into three methods, which are supervised, semi-supervised, and unsupervised anomaly detection [42]. Supervised anomaly detection contains both defect-free and defective samples, which are labeled in a training set. In this case, detection rates can be very high because all the training data is labeled. However, supervised anomaly detection is not the most efficient approach due to class imbalance in the data sets.

In semi-supervised anomaly detection, the training data set only includes labeled normal samples. This method is also called a one-class classification. The main idea is to learn and set a discriminative boundary that contains defect-free samples and considers any samples outside the boundary as anomalies. The method is very useful because it does not have to deal with obtaining a large number of defective samples, and the model is constructed just using normal samples. However, the anomaly detection accuracy of this method can be lower than that of supervised anomaly detection. The studies such as Refs. [43–46] apply different deep learning approaches in semi-supervised anomaly detection. Compared to the others, the semi-supervised anomaly detection presented in Ref. [46] shows more distinct decision boundaries around the outliers.

When labeled data are not available, unsupervised anomaly detection can be used. There are a few assumptions in it. The first assumption is that the majority of samples in the data set are normal (not defective). The defective instances are assumed to be rare in the data set. The second assumption is that the feature of anomalous instances should show noticeable deviations from that of standard instances in the data set. With these assumptions, the method learns the intrinsic characteristics of the data set to separate anomalies from the normal samples. For example, in Ref. [47], the authors demonstrate the anomaly detection based on unsupervised learning and deep GANs. The authors used only healthy data to train the GAN, and the constructed network is tested on both unseen healthy and abnormal data. Another example of unsupervised anomaly detection is seen in Ref. [48]. Here, the authors

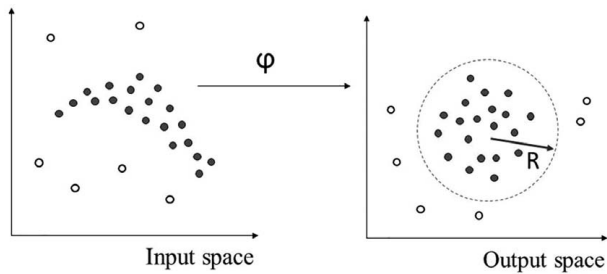


Fig. 3 By using the mapping, normal samples can fall within the hypersphere and defected samples fall outside the hypersphere (diagram based on work in Ref. [48])

introduce the method called deep support vector data description (DeepSVDD), which trains a deep neural network while jointly minimizing the volume of a hypersphere, which contains the network representations of the normal data. DeepSVDD performs this mapping and learns the neural network transformation with weights. The result of the algorithm shows that the data that falls outside the hypersphere can be classified as anomalies. Figure 3 shows the representative example of the data mapping from input space to output space.

3.2 Targeted Defect Detection. The anomaly detection described in Sec. 3.1 is desirable when classifying a data set into two groups, normal and defective samples. The anomaly group contains all types of defect samples in the data set. Compared to this anomaly detection method, the targeted defect detection method can be used to catch a specific type of defect by setting a specific defect as a target. A target defect is first identified, and the location of the target defect in an image is determined. This is done using supervised learning since the specific target should be labeled as the defect. An extensive amount of defective samples should be used to train the model. The type of defect and the amount of data acquisition affect the complexity of computations [49]. For clarification, we only consider detecting one specific defect as a targeted defect in this section. Detecting and identifying multiple defects is discussed in Sec. 3.3.

Compared to supervised anomaly detection, which only detects anomalies in data after training on the good samples, targeted defect detection trains a model or a deep neural network on the defected samples. The data acquisition process can be time consuming due to the need for substantial defective samples. For example, in a normal industrial setting and production line, the number of defect-free samples usually outnumbers the defective samples by a huge factor. So acquiring a large number of a specific defect sample can take time, and such a scenario might not be ideal. However, it works well when the defect data are easily available. Crack detection in pavements can be a good representative example [50], where CNN is used to detect pavement cracks.

3.3 Concurrent Identification of Multiple Defects. Surface defects can be of a variety of forms such as scratch, crack, inclusion, spots, dents, holes, and many more. The surface under inspection may contain one or more of these multiple defects. To identify the cause of the multiple defects on the surface, one needs first to identify them all. A simple surface anomaly detection does not work in this case as it does not classify defects. Targeted defect detection can be performed for each of the multiple defects. However, the targeted defect detection approach will not be an efficient one as it requires separately trained deep learning models for each defect. Also, it might fail if there is an interaction between the defects. For example, there might be a crack passing over a dent of a defective surface. So a better method to identify the multiple defects is necessary.

The approach for concurrent identification of multiple defects consists of a single deep learning architecture to identify the type

of defect on the surface under consideration. It can be done in a single step or can be divided into two steps: (1) identify if the surface has a defect and (2) classify the type of defect. The multiple defects identified in this approach consists of only the defects that are known. Each of these defects can be labeled manually or automatically. But they need to be labeled with the known categories of surface defects. So this type of approach is possible only using supervised and semi-supervised learning. The labeled images are used to train the deep learning model to identify the known defects. Training should include surface samples with multiple defects and defect interactions for the model to predict the defects concurrently. The recent work done in the concurrent identification of multiple defects is marked in Table 1.

Concurrent identification of multiple defects problems has been studied in Ref. [34]. The entire system architecture is divided into four stages, (1) anomaly detection, (2) filtering false anomaly, (3) clustering defect pixels, and (4) defect classification. It follows the two-step model of anomaly detection and defects classification, with the added stages of filtering and clustering. The diagram of the overall method with two convolutional neural networks is shown in Fig. 4. Here, the ResNet101 CNN is used for defect classification. The known defects are labeled using color codes, and supervised learning is performed. The six types of defects classified using the deep learning model for the DAGM2007 data set are shown in Fig. 5.

The limitation of the concurrent identification approach is the need for labeled defect data for training. Some defects are rare and do not have enough data or are not labeled. This can happen when a new manufacturing process is being adapted. In such cases, the defects may not be detected by this approach. But since these methods can detect defects concurrently, it allows the detection of defect interactions. This might enable us to detect interdependence between multiple surface defects. There is a need for studying the domain of using concurrent defect detection to detect surface defect relations.

3.4 Defect-Type Clustering. In Sec. 3.3, we discussed deep learning approaches to detect multiple known defects concurrently. In specific scenarios, all the surface defects may not be known. For example, if there are a few rare defects or infinitely many different types of defects, or if we are dealing with a new process resulting in a novel defect. The previous supervised or semi-supervised method cannot guarantee the correct detection of all the surface defects. This emphasizes a need for an unsupervised method for concurrent multiple defect detection.

The unsupervised method for multiple defect detection works by clustering similar types of defects. First, the anomalies on the surface are detected without any classification of the defect-type. The deep learning model takes all the defective surface samples and looks for similar defects in an unsupervised manner. The unsupervised deep model learns the characteristics of the surface defects, such as the shape, size, color, and more. These defect types are

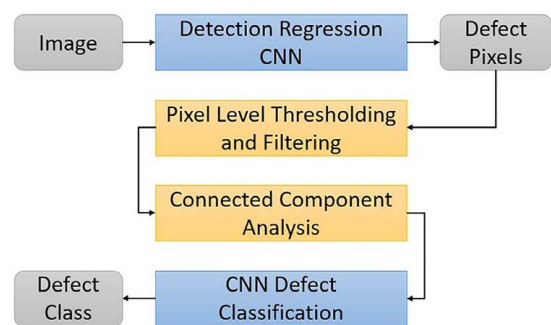


Fig. 4 An approach to concurrent identification method with two CNNs for anomaly detection and defect classification for multiple defects (diagram based on work in Ref. [34]).

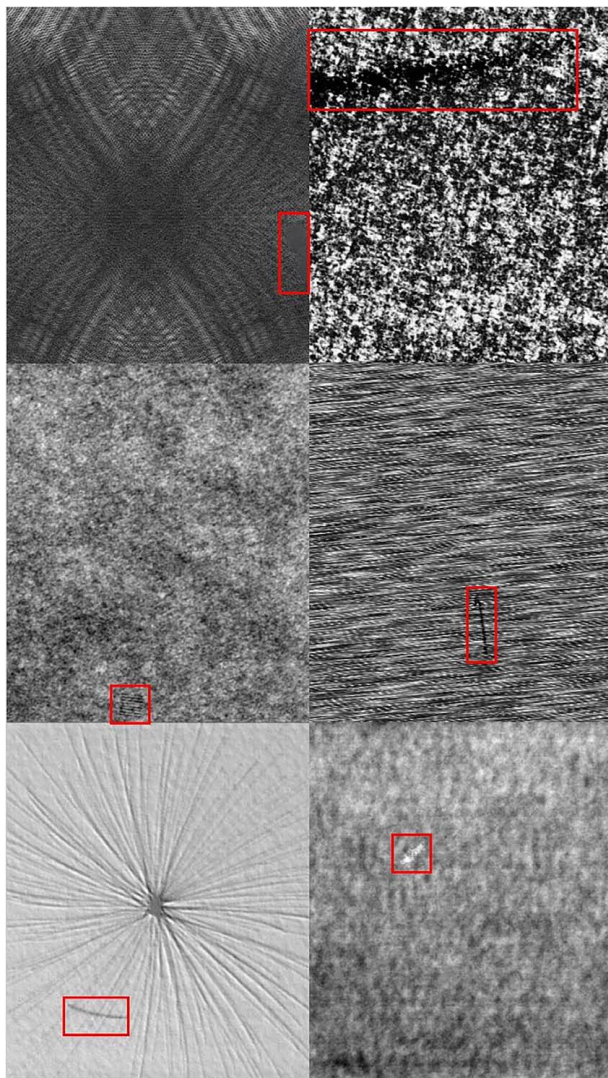


Fig. 5 Concurrent identification of six types of defects in the DAGM2007 data set [31] by using the deep learning model.

clustered by the model and reported to the user as type1, type2, etc. until all the defect types are classified. The advantage of such an approach is that there is no need for previous knowledge of the defect types or the labeling of the defect samples as the process is unsupervised. The recent research done in unsupervised defect clustering for defect classification is marked in Table 1.

Researchers have discussed a similar unsupervised approach in their survey on visual inspection of steel products [115]. The work is limited to self-organizing map ANN to classify multiple defects on the steel surface and does not discuss deep neural network-based methods for visual inspection. Researchers have utilized the unsupervised approach in defect classification [116]. They utilized two autoencoders and a Softmax probability classification layer in their deep learning model. The autoencoders are always trained in an unsupervised manner, but they train the Softmax layer in a supervised manner for steel surface defect classification. The different types of defects they considered are from the NEU steel surface database. If they had trained the Softmax layer using an unsupervised approach; it would be a defect clustering approach.

Currently, the work on unsupervised defect classification is minimal. One possible approach is defect-type clustering. It will enhance the surface defect classification approach and will enable us to deal with an unlimited type of defects. It will also remove the cumbersome process of defect type labeling for all training

samples. Researchers should explore this area for future research in deep learning for surface defect detection.

4 Learning-Based Classification

In this section, we divide learning-based approaches into supervised, semi-supervised, and unsupervised approaches. This division is motivated by the specific constraints faced by researchers in this area. Ideally, learning-based approaches perform best when a large data set is provided. Specifically, supervised approaches perform well when the data set is well balanced with sufficient examples for each class. There are several methods in the literature that use deep neural networks on existing data sets for defect localization, classification, and registration. For a specific application, if there is an existing data set, then supervised methods are leveraged. Section 4.1 briefly describes the supervised approaches.

A common issue with the application of deep learning for defect detection is the difficulty of obtaining a large data set crafted for the problem at hand. In particular, generating a labeled data set is either expensive and/or time consuming, and it is especially so due to how rare the defects are. Another issue is that the majority of deep learning methods are geared toward image classification or ROI specification. In defect registration, the defect region needs to be outlined as well as classified. This requirement is handled by mixing supervised approaches with unsupervised approaches in the learning pipeline. Section 4.2 briefly describes the semi-supervised and unsupervised approaches.

These issues have colored the approaches taken by many of the researchers in the field and have motivated them to invent novel methods to overcome the challenges of data sparsity, the intraclass variance between defect types, and the need for defect registration. The common techniques involve modifying the structure of CNN, incorporating specialized feature extraction, using transfer learning, and data augmentation.

Data augmentation is a general technique applicable to both supervised and unsupervised methods, which alleviates the problem of data sparsity. Typical operations include shifting/rotating images [75–108] and cutting up image patches into different sizes/scales. This allows features at a different scale to be included in the data set as individual samples and captures textural cues at different spatial scales [79]. Global noise is added to positive samples and includes them as negative samples. This increases sensitivity to localized features [79]. In Ref. [77], the defect is superimposed over defect-free samples. The superimposed defect is varied in terms of size, shape, and background color. Salt and pepper noise, Gaussian blur, Poisson noise, and motion blur are added in Ref. [65]. Data augmentation may skew the data and must be carefully used. We discuss each class of learning-based approaches hereon and how they use one or more of these techniques.

4.1 Supervised. Supervised methods requires large data sets to train effectively. Some of the data sets that supervised learning methods use are DAGM2007, Road-crack data set [117], Rail-road data set [118], fabric data set [119], silicon steel strips data set [120], and rail defect data set [70].

Supervised methods differ in how the deep neural networks are structured and the nature of feature extraction and classification. For example, in Ref. [75], it is claimed that the composition of kernels has more effect on the results than the number of layers after a certain number of layers. It also uses max pooling to be robust to small defect location changes in features. For surface inspection, it is necessary to determine the size of a sample image that is large enough to express small-sized defects as well as textures [75]. The research presented in Ref. [78] employs ROI pooling where the purpose is to perform max pooling to convert the features inside any proposals into vectors with a fixed size (e.g., 7×7). The specific operation of ROI pooling is shown in Fig. 6. The region proposals with different sizes are divided into

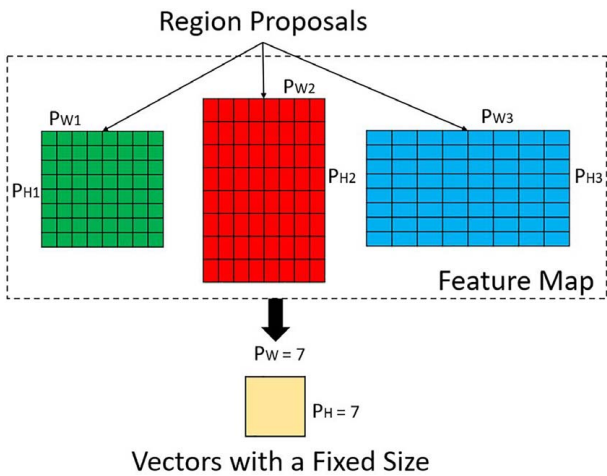


Fig. 6 Region of interest pooling (diagram based on work in Ref. [78])

equal-sized sections, such as 7×7 ; then, the max value in each section is output, and fixed-size vectors can be obtained.

In Ref. [54], a method is presented describing the merit of using shallow CNN networks (7.5M parameters) for anomaly detection. The premise is that the underlying defect structures and diversity of patterns are limited in their domain (10 defect classes, 100 defect samples per class). In light of this, the authors' use of shallow CNNs for defect detection is investigated. It evaluates whether shallower CNN architectures with fewer parameters can be used for automated visual inspection of surface anomalies while retaining a high classification accuracy. In the study, full-size images (as opposed to patches) are used, and only negative samples are used for training. As the negative samples also contain pixels corresponding to the defect-free region, the claim is that there is no need for full-size samples of both defect and defect-free samples.

An eleven layered CNN for classification and detection on the DAGM data set is presented in Ref. [66]. It consists of joint detection CNN architecture, which contains two major parts: the global frame classification part and the subframe detection part. The global frame classification part learns to classify the image samples into the correct class based on their background texture features. The subframe detection part is developed to decide whether each of the samples contains defective regions or not based on the output of the first part. The two parts are quite similar in architecture, and they are strung together for the defect detection forming the whole network.

MobileNet-SSD [36] is used to improve the real-time performance of deep learning under limited hardware conditions. This network can reduce the number of parameters without sacrificing accuracy. Previous studies have shown that MobileNet only needs 1/33 of the parameters of VGG-16 to achieve the same classification accuracy in ImageNet-1000 classification tasks. SSD network is a regression model, which uses features of different convolution layers to classify regression and boundary box regression. The model solves the conflict between translation invariance and variability and achieves good detection precision and speed. The complete model contains four parts: the input layer for importing the target image, the MobileNet base net for extracting image features, the SSD for classification regression and bounded box regression, and the output layer for exporting the detection results.

In Ref. [109], periodic defects like roll marks on hot-rolled steel plates are detected using a periodical defect detection method based on a CNN and LSTM according to the strong time-sequenced characteristics of such defects. Roll mark defects are not well detected because of the greatly different morphological features of roll marks on different batches of hot rolled steel. The traditional CNN classifies defects by extracted morphological features. Therefore, CNN can easily misclassify roll marks due to their unfixed morphological features. Consequently, the classification accuracy is not high. However, as roll mark defects have strong periodicity, their time-sequenced characteristics are suitable for handling by LSTM. Figure 7 shows the overall flow of CNN + LSTM. The features were extracted from the samples through CNN to obtain

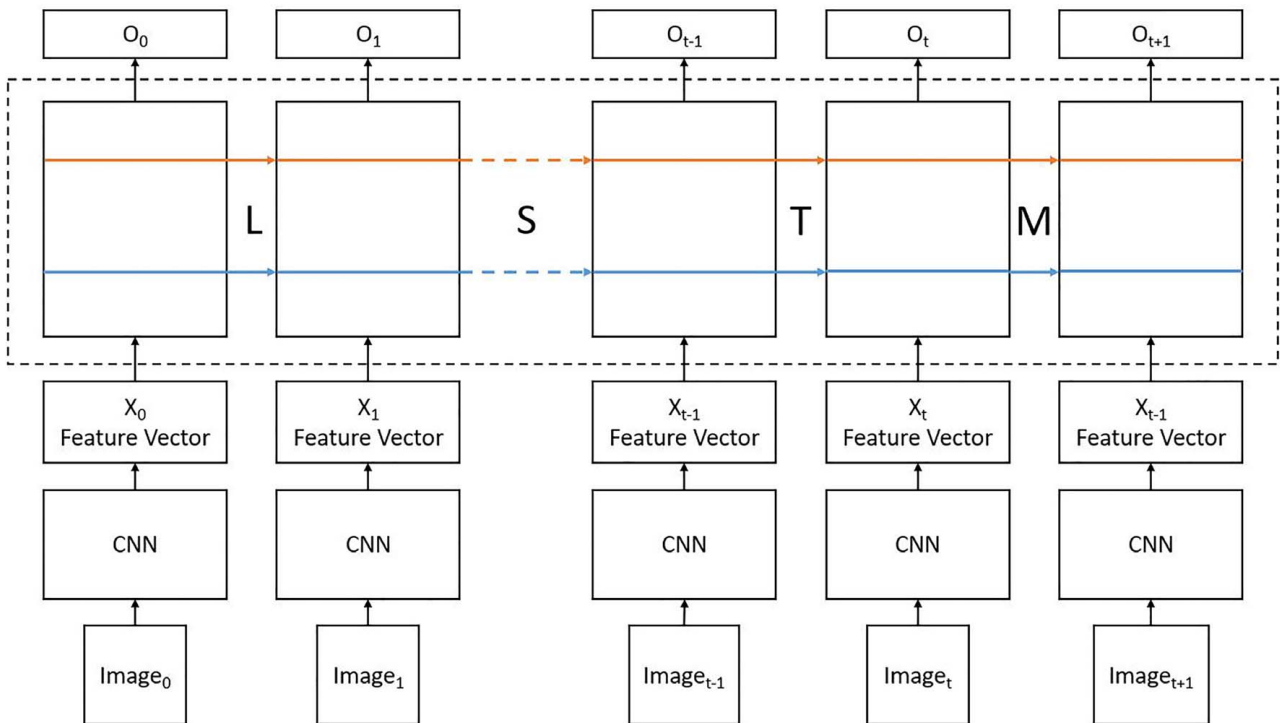


Fig. 7 Flowchart of the convolutional neural network and long short-term memory (CNN + LSTM) detection method (diagram based on work in Ref. [109])

their corresponding feature vectors. Then, the feature vectors were fed into the LSTM in a time sequence, and the outputs O are the recognition results.

Another aspect that differentiates supervised methods is the nature of feature extraction and classification. When the defects are of regular or predictable shapes, it is beneficial to use standard computer vision like object detection to identify the ROI [63]. In more complex defect types, more advanced preprocessing steps can be performed. Reference [76] generates candidate ROI as a preprocessing step before further processing.

Autoencoders are used for defect detection and CNN for defect classification for metallic surface defects in Ref. [14] and steel surface detection in Ref. [116]. The encoder-decoder network in Ref. [14] is based on the CASAE architecture consists of two levels of AE networks. An encoder network is a unit through which the input image is transformed into a multidimensional feature array for feature extraction and identification. The multidimensional feature array contains rich semantic information. On the other hand, a decoder network fine-tunes the pixel-level labels by merging the context information from the feature maps learned in all of the middle layers, as mentioned in Ref. [14]. The decoder network can further use an up-sampling operation to make sure that the final output is of the same size as the input image. Metallic surface defects are essentially local anomalies; hence, the actual defects and the background textures have different feature representations. The AE network is hence used to learn the representation of these local anomalies and find the common features between different defects. This problem of metallic surface defect detection is therefore turned into an object segmentation problem. The input defect image is transformed into a pixel-wise prediction mask with the encoder-decoder architecture, as mentioned earlier. The AE network produces a final prediction mask, which is the defect probability map used to detect anomalies. The probability map is the input to a CNN for classification [14].

4.2 Semi-Supervised and Unsupervised. Due to the data-sparsity problem, semi-supervised and unsupervised approaches tend to exploit transfer learning, data augmentation, and preprocessing. The research presented in Ref. [64] employs a novel data augmentation technique that increases the number of defect samples by cropping important regions of the defect image. Transfer learning is employed in Ref. [59], where a pretrained Decaf (deep CNN) is used as a feature extractor. A similar transfer learning approach is used in Ref. [108], where weights from AlexNet are used. The classification output layer of AlexNet is replaced with a randomly initialized two-class (i.e., defect/defect-free) classification layer for training. Reference [59] utilizes a pretrained deep learning network as an atomic building block for feature extraction. A pretrained SqueezeNet is used in Ref. [53]. In Ref. [83], defects are detected without relying on the labeled data.

With only a few reference images of defects, their method trains a deep autoencoder with augmented defect images to produce a defect descriptor. During testing, the descriptors of the test images are computed and compared against the defect descriptors. A similarity score is computed for the pair that indicates if a defect is present. In Ref. [79], a similar approach is used except CNN to generate the feature descriptors. The data-sparsity problem is addressed by research presented in Ref. [85] by using only defect-free samples to generate a discriminative representation.

The research presented in Ref. [77] augments the data using GAN and then uses a pixel-based CNN for classification. When the defect classes become too many for relative to the available samples for each class or new and unpredictable defect classes occur during production, only anomaly detection is a viable option [79]. It uses a feature space representation of an ideal part and a test part to compute a similarity metric. When the similarity metric is low, an anomaly is detected. An automatic data labeling method is presented in Ref. [65], where the data are labeled by extracting defect regions while also considering their relative scales.

5 Classification Based on Architectures Used for Defect Localization and Classification

In this section, we have identified different types of system architectures used to localize the defect and classify them into specific classes based on the application requirements (see Sec. 3). Each architecture is different because it trains the network to identify and localize defects in the image and classify them into defect classes. The input of each architecture is an image (with or without defect), and the output is the defect location and defect class.

5.1 Image Classification-Based Localization: Architecture

1. This architecture (see Fig. 8) takes the entire image as input and only outputs if the image contains a defect or not. In other words, the defect's location is not specified; however, an entire image can be considered a defect. This architecture is commonly known as image classification [62] and is one of the most common tasks performed by a deep learning network. The most common application that uses this architecture is anomaly detection. The amount of data needed to train a neural network in this architecture is comparatively low, and the network can be trained in a semi-supervised or unsupervised manner. However, the downside of this architecture is that it cannot localize the defect on the image, which can be essential for a large number of product inspection applications (see Fig. 9).

The approach presented in Ref. [52] is a classic case of this architecture where a CNN network is trained for deflectometric inspection of specular surfaces. Authors in Ref. [53] use CNN to

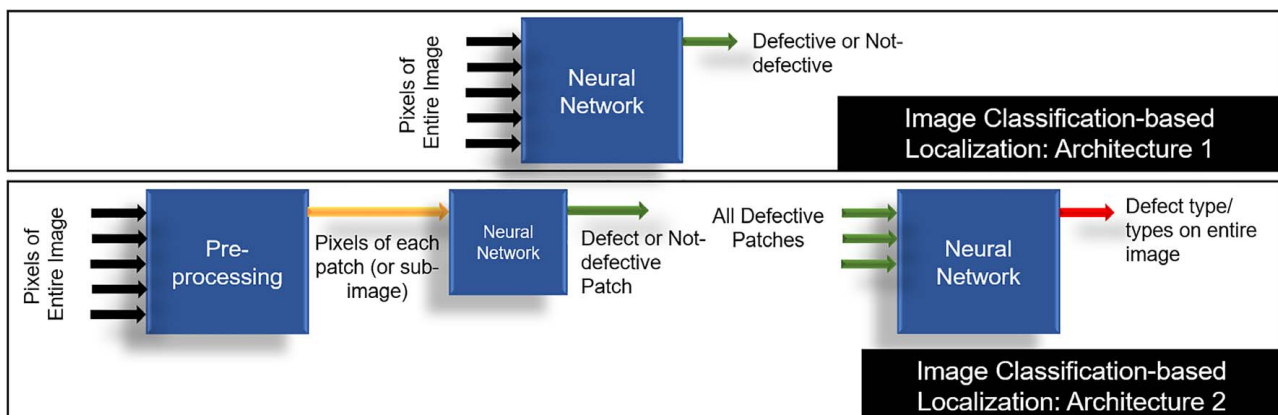


Fig. 8 Defect detection architectures using image classification-based defect localization

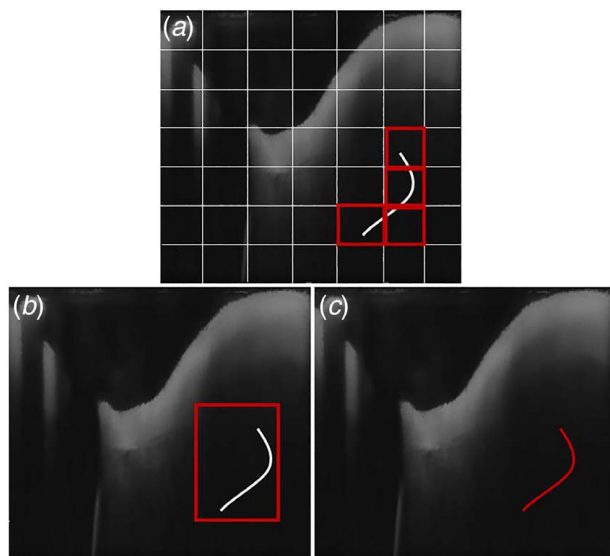


Fig. 9 Defect detection using three approaches: (a) image classification-based localization: Architecture-1, (b) object detection-based localization: Architecture-1, and (c) image classification-based localization: Architecture-2 (illustration based on work in Ref. [62])

achieve fast and accurate steel surface defect classification into crazing, inclusion, patches, pitted surface, rolled-in scale, and scratches. Reference [74] uses CNN for classifying weld images into good quality welds, over spatter, porosity, and undercut.

Authors in Ref. [52] have used the same architecture. However, they modify it to improve the accuracy of defect classification with low availability of labeled data. The same image is passed through three different CNN for feature extraction, which is later merged by a feature-fusion module and passed through a CNN-based classifier to determine the class of defect.

5.2 Image Classification-Based Localization: Architecture 2. This architecture (see Fig. 8) uses a preprocessing unit that segments the entire image into small images that are called a patch. One of the most popular preprocessing operations is performed using a sliding window. The usual approach is to divide the image using a sliding window into smaller patches. Each patch is then passed through a neural network and labeled with a discrete label suggesting defective or defect free. Finally, the use of a postprocessing step is required, which combines all the defective patch provides a location of the defect on the image.

Apart from the design of the CNN architecture, this approach is relatively simple for anomaly detection and localization. However, selecting the right size of the sliding window is challenging and has to be manually tuned based on the type and the size of defects. Selecting the size of the window to be too large reduces the accuracy of the defect localization on the image and having the window size small increases the signal-to-noise ratio and reduces the classification accuracy.

In this architecture, defect classification is performed depending on the application in one of two ways: (a) In applications where the defects are small and fit, each patch size can be classified into

various defect classes by the same neural network used for anomaly detection. (b) In application, where defects are spread over multiple patches needs to be combined and later on passed through a second neural network for classification.

One of the examples of this architecture is presented in Ref. [64], where authors present a new CNN named small data-driven CNN (SDD-CNN) and tested in for defect detection on roller surfaces. Each of the images of the roller was divided into patches, which were used for training and evaluation of the classifier. The end result is that each patch was classified into a separate class of defects. The authors compare SDD-CNN with the original CNN models, and the new SDD-CNN is better in terms of convergence speed, training time, and classification accuracy. Similarly, Ref. [80] divides the image into patches and uses Fisher criterion-based deep learning method for detecting defects in fabrics. The approach presented in Ref. [66] segments the image in patches, which are first classified based on texture and later passed into another CNN for anomaly detection. Finally, Ref. [58] uses this architecture for detecting cracks, which results in poor defect localization due to constant patch size.

Few researchers have used traditional image processing approaches to extract the region of interest from the entire image. Here, a region of interest can be the object of interest or area where the defect is likely to occur. For example, Ref. [69] uses image segmentation by the Sobel edge filter and a binary threshold for extracting the region of interest on the copper clad lamination surface. The approach presented in Ref. [63] uses the image processing method of image contour query to localize the location of the screw head in the image before passing through a CNN (LeNet5) for classifying the class of defect on screw heads.

The approach presented in Ref. [59] first builds a classifier on the features of image patches, where the features are transferred from a pretrained deep learning network. This step of feature extraction from a previously trained network significantly reduces the amount of data needed for training. Now, the accuracy of defect localization is achieved by a pixel-wise prediction by convolving the trained classifier over the input image. For each defect class, a heat map is obtained by iteratively adding the probabilities pixel-wise. This is combined in a later stage to identify the entire defect in the image.

5.3 Pixel-Based Localization. Pixel-based localization (see Fig. 10) is at another end of the spectrum as compared to this architecture-1 discussed earlier. The input to this architecture is the entire image, and the output is the image of the same size with probabilities of defect on each pixel. This allows the architecture to accurately (at pixel-level) locate the defect on the image. Moreover, the same network is used to classify the type of defect for each pixel.

The technique assigns a semantic class to each pixel in an image. In the defect detection domain, the class represents if a pixel belongs to a defect or not. Some methods also account for which type of defect the pixel belongs to Ref. [34]. For instance, a pixel could belong to a misalignment, crack, abraded surface, etc. A deep learning model is trained using images with such semantic labels. The trained model is queried to obtain a score for each individual pixel. This score could be the probability with which that particular pixel belongs to a defect or a type of defect.

The work presented in Ref. [86] uses CrackNet to detect the pixels corresponding to defects. Input to the CrackNet is the aggregation of all feature vectors corresponding to an image in

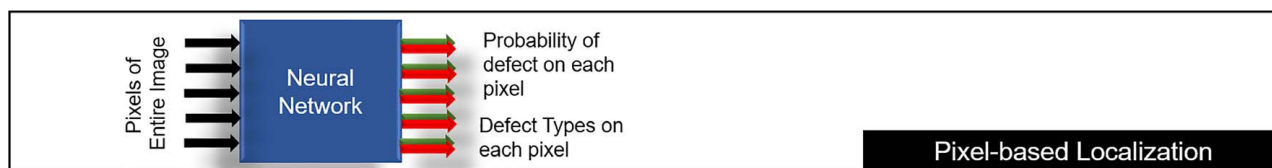


Fig. 10 Defect detection architecture using pixel-based defect localization

the training set. Pixel-perfect accuracy is achieved by keeping the spatial size of images invariant throughout. Individual pixel is compared with its neighbors, and a final score to each pixel is assigned as an output. The benefits of image classification-based localization: architecture-2 architecture can be summarized as follows:

- (1) Downsizing of the original image can be avoided. Unlike CNN [82], which uses pooling layers to downsample the image, networks like CrackNet preserves the spatial invariance.
- (2) Segmentation techniques failed to detect crack width accurately as detection is at the block level instead of pixel level.
- (3) Localization can be achieved at pixel-perfect accuracy as opposed to accuracy given by bounding box or set of patches.
- (4) Defect detection problem can be solved as an anomaly detection problem by only using a few numbers of defect-free samples, which are readily available in the industry.

The method proposed in Ref. [114] uses a layered architecture where patches are first used to obtain a representation of cracks in the image using patterns in each patch. A local pattern predictor uses CNN to extract these discriminative features of the image. Then each pixel is categorized into the crack or the noncrack category using a small neighborhood around the pixel. The output of the method is a confidence map is used to obtain the crack areas. They predict the probability of each pixel in a patch to belong to a crack based on the pattern in the patch.

Pixel-based architecture is also used to learn semantic image features. This requires a low number of training data. A variant of transfer learning was proposed in Ref. [84]. The work aimed at detecting defects like scratch, missing washer/extra hole, and abrasion in printed circuit boards by posing it as an anomaly detection problem. The model extracts semantic features from images in an unsupervised manner. Normal features on PCB form a cluster, and abnormal features form a separate cluster. A multimodal Gaussian pyramid scheme with convolutional denoising autoencoder (CDAE) network at each level was proposed for defect detection on textured surfaces using patches in Ref. [85]. The distribution of patterns in reference images are learned. A defect is present if these patterns are different. Multiscale CDAE is training that includes image processing, patch extraction, and model training. Textural image patches at different resolution scales can be reconstructed with a convolution denoising AE in each pyramid layer.

5.4 Object Detection-Based Localization: Architecture 1. Object detection-based localization architecture 1 (see Fig. 11) treats defect detection problem as an object detection problem, where the goal is to identify the location of the object using a bounding box and decide the object type. For the inspection domain, the objects are defects. This type of architecture uses a region proposal network (R-CNN) [121] to determine the regions

of interest, which is later used by a fully connected CNN for classification.

Unlike, classification-based approaches (see Secs. 5.1 and 5.2) object detection-based approaches does not require the sliding window size to be modified on case-to-case basis. Moreover, object detection-based approaches predict the location of the defect with higher precision as it is done using a bounding box and not with a fixed-size window. This method is popular among the real-world (or real-time) application as it uses the image as a whole, and CNN only has to run once compared to a sliding window where the CNN has to run on each patch independently, which makes the process computationally expensive [62].

Approach presented in Ref. [60] presents a classical case of object detection-based localization architecture 1. First, object detection-based segmentation is used to detect insulators that are connected to the catenary wire used for the traction power supply system in the electrified railway. Once the insulator is segmented, the deep learning network uses DMC and DDAE for defect detection.

A few of the popular CNN architecture that belongs to this architecture involves fast R-CNN [122], faster R-CNN [123], which is a combination of region proposal network (RPN), and Fast R-CNN. The approach presented in Ref. [100] uses R-CNN for defect localization on PCB, which is followed by a full CNN for defects type classification. Similarly, Ref. [113] uses an object detector network for localizing the weld on the image. Later on, the surface properties of the weld seam using a DenseNet-121.

Authors in Ref. [65] use Faster R-CNN for detecting surface defects on Wheel Hub. RPN, which is used to generate the proposals, and FAST R-CNN is used to locate the object accurately. The approach presented in Ref. [30] proposes a method based on Faster-RCNN and feature fusion for defect detection. The performance of the algorithm is tested on the CAGR and NEU database.

5.5 Object Detection-Based Localization Architecture 2. Object detection-based localization architecture 2 (see Fig. 11) is an extension of object detection-based localization architecture 2 where the same architecture is used for localization and defect classification. For example, the approach presented in Ref. [78] proposes a new multiscale defect detection network that added several fully connected layers at the end of the faster-RCNN for defect type classification. This network is tested on identifying defects on aluminum profile surfaces. Training data required for this type of architecture is large as it needs to perform localization and classification simultaneously.

6 Classification of Existing Literature

Overview. We summarize the existing literature according to the classification discussed in Secs. 3–5 in Table 1. Additional information, such as the type of neural network used and the

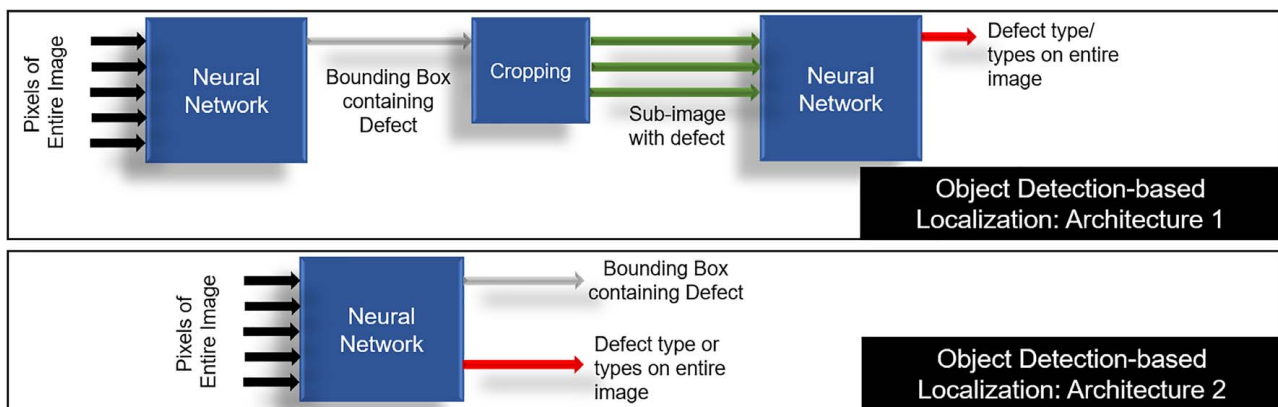


Fig. 11 Defect detection architectures using object detection-based defect localization

Table 1 List of the recent research publications in deep learning defect detection along with the classifications

| ID | Classification | | | Comments |
|-------|--|-----------------------------------|--|--|
| | Classification based on requirements and context (A) | Learning-based classification (B) | Classifications based on architectures used for defect localization and classification (C) | |
| | | | | |
| [51] | A.1 | B.1 | C.1 | Motor magnetic tile; FFCNN |
| [52] | A.1 | B.1 | C.1 | Specular surfaces; CNN |
| [34] | A.3 | B.1 | C.3 | Capacitor, DAGM, AigleRN; ResNet 101 |
| [53] | A.3 | B.1 | C.2 | NEU steel surface; SqueezeNet |
| [54] | A.1 | B.1 | C.3 | DAGM data set; shallow CNN |
| [30] | A.3 | B.1 | C.5 | GAPR texture data set; faster RCNN and ResNet |
| [29] | A.3 | B.1 | C.3 | DAGM data set; FCN |
| [55] | A.1 | B.1 | C.5 | DAGM data set, screw, and gasket; CNN |
| [56] | A.1 | B.2 | C.1 | Railway Rail; CNN |
| [57] | A.1 | B.1 | C.1 | Mangosteen fruit; CNN |
| [58] | A.2 | B.1 | C.4 | Concrete surface; CNN |
| [59] | A.3 | B.1 | C.1 | Misc.; Decaf CNN |
| [60] | A.2 | B.2 | C.4 | Catenary wire insulator defects; faster R-CNN |
| [61] | A.3 | B.1 | C.2 | custom data set; AlexNet with SVM |
| [62] | A.3 | B.1 | C.4 | Crankshaft assembly inspection; CNN |
| [63] | A.3 | B.1 | C.1 | Micro-defect on screw surface; LeNet5 |
| [64] | A.3 | B.2 | C.1 | Defects on roller surfaces; SDD-CNN |
| [65] | A.3 | B.1 | C.4 | Surface defects on wheel hubs; faster R-CNN |
| [66] | A.1 | B.1 | C.2 | DAGM data set; CNN |
| [67] | A.1 | B.1 | C.2 | DAGM data set; FCN |
| [68] | A.3 | B.2 | C.1 | solar panel surface and wood texture; GAN |
| [69] | A.1 | B.2 | C.2 | copper clad lamination; CNN |
| [70] | A.3 | B.1 | C.1 | Rail surface defects; DCNN |
| [71] | A.1 | B.2 | C.1 | Surface defects; GAN |
| [72] | A.3 | B.1 | C.5 | Steel strips; faster-RCNN |
| [73] | A.2 | B.1 | C.3 | Surface crack on plastics electronic commutators; LNET CNN |
| [54] | A.1 | B.1 | C.3 | DAGM data set; shallow CNN |
| [74] | A.3 | B.1 | C.1 | Welding; CNN |
| [75] | A.3 | B.1 | C.1 | Texture; CNN |
| [76] | A.3 | B.1 | C.2 | Misc.; CNN |
| [77] | A.1 | B.1 | C.1 | LCD glass cover; GAN |
| [65] | A.3 | B.1 | C.2 | Wheel hub; CNN |
| [78] | A.3 | B.1 | C.5 | Aluminum profile; Faster R-CNN |
| [53] | A.3 | B.2 | C.1 | Aluminum profile; CNN |
| [69] | A.1 | B.2 | C.1 | Copper surface; CNN |
| [79] | A.1 | B.2 | C.1 | Texture; CNN |
| [80] | A.2 | B.2 | C.2 | Textured fabrics; fisher criteria segmentation, CNN |
| [81] | A.3 | B.1 | C.1 | Metal surface; CNN, SVM |
| [82] | A.1 | B.1 | C.5 | Aluminum welding; CNN |
| [83] | A.1 | B.2 | C.1 | Misc.; deep AutoEncoder |
| [84] | A.1 | B.1 | C.4 | Printed circuit boards; transfer learning |
| [85] | A.1 | B.2 | C.1 | Textured surfaces; convolution denoising autoencoder |
| [50] | A.2 | B.1 | C.1 | Pavement crack analysis; CNN |
| [86] | A.2 | B.1 | C.3 | Pavement cracks; CrackNet |
| [58] | A.1 | B.1 | C.1 | Concrete surface; CNN |
| [87] | A.3 | B.1 | C.2 | Rolled steel strips; max-pooling CNN |
| [88] | A.1 | B.1 | C.2 | DAGM data set; custom CNN architectures |
| [89] | A.1 | B.1 | C.4 | Leather; R-CNN |
| [90] | A.3 | B.1 | C.4 | Steel surface; SDD, ResNet |
| [91] | A.3 | B.1 | C.4 | Steel surface; SDD, ResNet |
| [92] | A.3 | B.1 | C.4 | Rail surface; CNN |
| [93] | A.1 | B.1 | C.1 | Wafer surface |
| [94] | A.3 | B.1 | C.1 | Rail surface; DCNN |
| [95] | A.3 | B.1 | C.3 | Steel surface; NEU-DET data set |
| [96] | A.3 | B.1 | C.3 | DAGM, NEU-seg, MT_defect, Road-defect data sets |
| [97] | A.2 | B.1 | C.1 | Nuclear fuel rods; CNN |
| [98] | A.1 | B.2 | C.3 | Misc.; deep autoencoders |
| [99] | A.2 | B.1 | C.3 | Steel surface; GAN |
| [100] | A.3 | B.1 | C.4 | PCB board errors; R-CNN |
| [101] | A.3 | B.1 | C.5 | Metal AM laser powder bed defects; CNN |
| [102] | A.3 | B.1 | C.2 | Automotive engine precision parts; PartsNet |
| [103] | A.3 | B.1 | C.4 | Fasteners on the catenary device; SDD, YOLO |
| [104] | A.2 | B.1 | C.3 | Crack detection; SDD |
| [105] | A.3 | B.1 | C.2 | Solar cell surface; CNN |
| [106] | A.3 | B.1 | C.1 | DAGM 2007 data set; CNN |
| [107] | A.1 | B.1 | C.4 | Rail defect detection; Edge detection, CNN |
| [108] | A.3 | B.2 | C.1 | Infrastructure inspection; AlexNet |
| [14] | A.3 | B.1 | C.3 | Metallic surface; CNN |

Table 1 Continued

| ID | Classification | | | Comments |
|-------|--|-----------------------------------|--|---|
| | Classification based on application requirements and context (A) | Learning-based classification (B) | Classifications based on architectures used for defect localization and classification (C) | |
| | | | | |
| [109] | A.1 | B.1 | C.1 | Hot-rolled steel plates; CNN+LSTM |
| [110] | A.2 | B.1 | C.4 | COCO data set; ResNet & image pyramid CNN |
| [111] | A.3 | B.1 | C.4 | Powder bed fusion; ResNet, Faster-RCNN |
| [112] | A.3 | B.1 | C.5 | Metal AM errors; CNN |
| [113] | A.2 | B.1 | C.2 | Quality of friction stir weld; DenseNet-121 |
| [114] | A.2 | B.1 | C.2 | Bridge surface; local pattern predictor |

Note: Here, headers A, B, and C refer to the Secs. 3, 4, and 5, respectively. A.1, anomaly detection; A.2, targeted defect detection; A.3, concurrent identification of multiple defects; A.4, defect-type clustering; B.1, supervised; B.2, semi-supervised and unsupervised; C.1, image classification-based localization: Architecture 1; C.2, image classification-based localization: Architecture 2; C.3, pixel-based localization; C.4, object detection-based localization: Architecture-1; C.5, object detection-based localization: Architecture 2.

application focus of the research, is also mention for each literature. This will allow the readers to find and review the interesting research work easily.

6.1 Characterization of Deep Neural Network Configurations. Deep learning is analogous to the concept of DNN, which is a part of the ANN domain. DNN provides the advantage of avoiding feature engineering and predicting extremely complicated relationships provided enough training samples are provided. Like any learning technique, they are affected by overfitting. CNN is a type of DNN where convolutional layers are added to reduce the number of training parameters (weights and biases). CNN is particularly useful when dealing with image data as they have high dimensions (breadth pixels \times height pixels). However, CNN layers apply the convolution filter, which causes the loss of data. CNN is very useful in surface defect detection as the primary mode of input is images. Softmax layers are useful for classification networks. They are advantageous in defect detection tasks as they allow to convert the network results to the probability of each class in the classification network. LSTM provides the benefit of predicting the relationship between defects on the surface. For example, if a pore appears on the metal surface, it can develop into a crack. These kinds of dependencies can be predicted using LSTM.

There are multiple CNN-based networks that have produced impressive results over the past few years. A few examples are VGG net, ResNet CNN, SqueezeNet CNN, and FCN [29]. These networks primarily differ in their architecture and hence excel at specific tasks. As time progresses, these architectures have evolved to retain the core principles that worked for the class of problems they were targeted for. For example, ResNet is a deeper version of VGG. Besides the problem types they handle, the methods also vary in terms of the size and memory requirements for training. For example, SqueezeNet is a CNN architecture that has roughly the same accuracy as AlexNet. However, it requires 50 times fewer parameters and has a significantly small model size. This allows this method to be applied in hardware with limited memory or communication constraints. Other architectural differences include the balance between the number of parameters and the resulting accuracy. In SqueezeNet, convolutional filters are judiciously downsized from 3×3 to 1×1 in an attempt to conserve the parameter count budget. Another technique is to downsample the image late in the pipeline. This leads to convolutional layers with large activation maps and ultimately results in higher classification accuracy.

6.2 Characterization of Localization Approaches. Image classification-based localization architecture 1 uses a single or chain of deep learning models that take the image as an input and only output a binary decision of whether the defect is present in

the image. The process is known as image classification. Image classification-based localization-architecture 2 first preprocesses the image to divide it into patches. A popular approach known as the sliding window is often used to segment the image. Each patch is then passed through the network and labeled as defective or nondefective. Finally, a postprocessing step clusters all images and gives out the location of the defect on the image. If defects are small and easily fit a patch, detection and classification are done by the same network. If defects are large and spread over multiple patches, a separate network is used for classification. The pixel-based method or pixel-based localization uses the pixels of the entire image as an input to the neural net. Then, each pixel is first labeled as belonging to a defect or not and then classified according to defect class. Later all defect pixels are clustered, and their class and localization are determined. Object detection-based localization-architecture 1 treats the problem as an object detection problem where a bounding box is generated around the defect using a neural net, and then, the image is cropped to the box and passed through another network for classification and detection. Object detection-based localization-architecture 2 is an extension of the previous method where the same network is used for generating the bounding box and defect classification.

Image classification-based localization-architecture 1 is known for using fewer samples in anomaly detection literature. Unsupervised or semi-supervised learning can also be performed. However, defect localization or classification is not performed. Image classification based localization-architecture 2 is powerful since it detects and classifies and then localizes the defect. However, choosing the right size of the sliding window is a challenging task. The pixel-based method is accurate since it performs operations at a pixel level. However, the input vector dimension becomes huge as we are dealing with the entire image at once. Object detection based localization-architecture 1 uses the image as a whole, and the neural network has to run only once on the image. This approach is quite popular in real-time applications. Object detection-based localization-architecture 2 is an extension of architecture 1 in the same category and requires fewer data to perform localization and classification simultaneously.

7 Future Research Directions

The field of deep learning is changing rapidly. Recent advances are expected to impact the defect detection area as well. This section describes recent trends and future research directions.

Using deep learning with limited defect data: Often, people struggle to find an adequate amount of data for successfully deploying deep learning in defect detection applications. In the manufacturing or production line, the number of defect-free parts produced is much higher than the number of defective parts.

Therefore, the data with defects is inherently small. For a simple anomaly detection method that can train well on normal samples, the small defect data set is not a problem. However, for defect localization and classification, the size of the data set containing defects can become a challenge.

A possible method to solve this problem is data augmentation [124]. There are several techniques in data augmentation, such as geometric transformation (flipping, cropping, rotating), random erase, image mixing, feature space augmentation, and color space augmentation. Also, changing lighting conditions such as exposure or brightness is another technique to create more data. GAN or meta-learning can be used as well to create synthetic data. These techniques are used in the preprocessing data stage.

Explainability: When a defect detection method fails to find a defect or incorrectly identifies a defect in an acceptable part, users are interested in understanding why the system failed. Unfortunately, mostly deep learning methods use a complex architecture, and hence, it is difficult for humans to understand the decision-making process and provide a rationale for failure. This can become a challenge in deploying and improving system performance.

Recent work in deep learning is focused on improving the explainability of the observed system performance [125,126]. In addition, physics-based relationships can be established as ground truth and given to the model to ensure that prediction is consistent with physics-based models. Defect detection community will need the development of new techniques that can explain the decision-making by the deep learning architecture.

Transfer learning: Two different application domains may share defect patterns. For example, cracks in two different materials may share a similarity in morphology but may be different in terms of colors and sizes. Current approaches require users to train two different networks. It will be useful to transfer learning from a well-trained and tested network to another to expedite the training. Most current approaches do not effectively utilize transfer learning.

We believe that transfer learning can play a role in providing an appropriate seed for the weights and structure of the network in defect detection applications. Transfer learning [127] can allow the neural network to reuse the feature extractor portion of a previously trained network using existing large data sets and retrain only the classification functionality using specific data sets appropriate for different classification tasks. As mentioned in Ref. [107], transfer learning is a learning method that uses existing knowledge to solve problems in different but related fields. It relaxes two basic assumptions in traditional machine learning to migrate existing knowledge to deal with learning problems in the target area where there is only a small quantity of tagged sample data set. There is a need to develop a taxonomy of different defect detection applications and characterize what kind of learning can be transferred among these applications.

Finding balance between automatic feature detection and hand-crafted feature detection rules: Defect detection methods require reliable features to work well. Hand-crafted feature detection rules are reliable but require domain experts to define features. These features differ based on the paradigm used (e.g., statistical, pixel-structural, filter based, model based) [22]. When these features are tuned specifically for the application area, they work well and produce good defect detection. However, collecting data from domain experts are often a laborious and expensive process and requires significant programming effort. Moreover, new defects may appear after collecting data from domain experts. It becomes impractical to repeat the data collection process frequently. On the other hand, automatic feature extraction does not require manual feature extraction by experts but requires a large amount of data for automatic feature learning methods to be robust to noise, illumination, scale, and rotation changes. Furthermore, the data set needs to be balanced, containing a balanced number of samples for each class [115].

It appears that finding the right balance between hand-crafted feature detection rules and automated feature extraction methods might provide a balance between two approaches. Features that

are distinctive can be detected using methods such as bilateral filtering, Sobel filtering, Laplacian filtering, Canny, morphological operations, and thresholding can be applied for feature extraction [22,115,128] during a preprocessing step and can be fed into a deep learning network along with the image. The deep learning network can automatically extract additional features via a CNN layer [74,78]. Such a hybrid scheme can combine the strengths of two approaches.

Scale invariant defect detection: Object detection architectures presented in Secs. 5.4 and 5.5 have ability to accurately localize the defects in the image. However, it is susceptible to poor accuracy when using images of different scales and skewed proportions. The reason for this is that different layers of CNNs have the capability of different levels of abstraction and capture the different amount of structure from the patterns present in the images used for training. These learning of features take place by accounting for pixel-level information in an image. Having images of different scales changes the covariance around each pixel and thus making it hard to detect. Some applications may require imaging to be done from different distances, and therefore, defects might appear in widely varying sizes.

To train a robust CNN network, it is imperative to train the network to look for scale-invariant features. One of the popular networks that are resilient to scale is feature pyramid networks [129], which generates multi-scale feature maps. Also, using GANs that narrow representation differences between small and large objects can also be useful. Moreover, for scale-adaptive feature detection, it might be useful to combine scale distribution estimation [130], attentional mechanism [131], and knowledge graph [132] to detect defects of varying sizes.

Integration of physics-based reasoning: Deep learning is a statistical method capable of predicting accurate models from data. The learning of the model depends on the size of data and noise-to-signal ratio. Unreliable training data lead to poor performance and instability [133]. Surface defect detection uses images that are affected by a variety of conditions, such as lighting, exposure, and more. The images tend to have noise embedded in them. When such images are used to train the deep learning network, it can lead to overfitting the noise. This is because a deep learning network has the tendency to learn whatever it can. When such a noisy model is used to predict the anomalies and defect types, it can give false positives and wrong classifications.

Physics-based reasoning is based on deeper expert domain knowledge and is not data-driven. They do not rely on data availability. They have limited accuracy and cannot cover the entire regime due to the necessary simplifications in the physics model. By combining deep learning and physics-based reasoning, we can avoid noisy predictions with good accuracy and regime [133]. In surface defect detection, we can include physics-based reasoning as a penalty or evaluation function while training the deep neural network. It can also be included in the postprocessing step, and the false predictions can be avoided. Thus, even highly noisy surface image data can be used to train and predict the anomalies and defect types.

Avoiding overfitting: Deep learning models are trained by automatically tweaking a large number of parameters. When the number of data samples is small compared to the number of parameters, the model runs the risk of being overfitted. While training error may reduce, the model may not be accurate for novel data samples that were not in the training data set. The goal of learning is to produce a model that can generalize over a wide variety of input data.

The simplest way to solve the problem is to inject more data samples by using data augmentation techniques mentioned in Sec. 4. A fundamental technique is to modify the network structure. Dropout can be introduced to the network layers [106]. Dropout refers to turning off the output of a number of random neurons in a given layer during training. This helps the overall network be robust to minor data variations and forces the network to learn the underlying physics of the problem. Other techniques involve:

(a) monitoring validation accuracy and stop training as soon as validation loss begins to rise again. (b) Penalize high neuron weights in the model [105,106]. The intuition is that a smaller neuron weight allows for a more gradual change in a neuron's activation, thus regulating the response of the network to wild changes in the input. (c) Incorporate a pixel-wise loss function [104].

8 Conclusions

Deep learning is gaining popularity in the defect detection community. This article presents three different perspectives for examining the existing literature. The first perspective is based on identifying the scope of different detection problems based on application contexts and requirements. This perspective helps us define and understand different types of defect detection problems. The second perspective examines the literature from a machine learning perspective and explains why certain learning approaches are useful for certain kinds of problems. Finally, the system architecture perspective explains different types of approaches used to localize and classify defects from a system architecture point of view. We classify literature using these three perspectives. Image-based surface defect detection using deep learning is a fast emerging field and presents unique challenges compared to other image analysis and object detection problems. We also identify and present directions for future research.

Acknowledgment

This work is supported in part by National Science Foundation (Grant No. 1925084). Opinions expressed are those of the authors and do not necessarily reflect the opinions of the sponsor.

Conflict of Interest

There are no conflicts of interest.

References

- [1] Park, M., Jin, J. S., Au, S. L., Luo, S., and Cui, Y., 2009, "Automated Defect Inspection Systems by Pattern Recognition," *Int. J. Signal Process., Image Process. Pattern Recognit.*, **2**(2), pp. 31–42.
- [2] Tsa, D.-M., and Wu, S.-K., 2000, "Automated Surface Inspection Using Gabor Filters," *Int. J. Adv. Manuf. Technol.*, **16**(7), pp. 474–482.
- [3] Tsai, D.-M., and Huang, T.-Y., 2003, "Automated Surface Inspection for Statistical Textures," *Image Vis. Comput.*, **21**(4), pp. 307–323.
- [4] Samarawickrama, Y. C., and Wickramasinghe, C. D., 2017, "Matlab Based Automated Surface Defect Detection System for Ceramic Tiles Using Image Processing," 2017 6th National Conference on Technology and Management (NCTM), Malabe, Sri Lanka, Jan. 27, IEEE, pp. 34–39.
- [5] Elbehery, H., Hefnawy, A., and Elewa, M., 2005, "Surface Defects Detection for Ceramic Tiles Using Image Processing and Morphological Techniques," *Egyptian Inf. J.*, **6**(1), pp. 123–133.
- [6] Iivarinen, J., 2000, "Surface Defect Detection With Histogram-Based Texture Features," *Intelligent Robots and Computer Vision xix: Algorithms, Techniques, and Active Vision*, Boston, MA, Oct. 11, Vol. 4197, International Society for Optics and Photonics, pp. 140–145.
- [7] Jie, L., Siwei, L., Qingyong, L., Hanqing, Z., and Shengwei, R., 2009, "Real-Time Rail Head Surface Defect Detection: A Geometrical Approach," 2009 IEEE International Symposium on Industrial Electronics, Seoul, South Korea, July 5–8, IEEE, pp. 769–774.
- [8] Huang, Y., Qiu, C., Guo, Y., Wang, X., and Yuan, K., 2018, "Surface Defect Saliency of Magnetic Tile," 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE), Munich, Germany, Aug. 20–24, pp. 612–617.
- [9] Jia, H., Murphey, Y. L., Shi, J., and Chang, T.-S., 2004, "An Intelligent Real-Time Vision System for Surface Defect Detection," *Proceedings of the 17th International Conference on Pattern Recognition*, 2004. ICPR 2004, Cambridge, UK, Aug. 23–26, Vol. 3, IEEE, pp. 239–242.
- [10] Xue-Wu, Z., Yan-Qiong, D., Yan-Yun, L., Ai-Ye, S., and Rui-Yu, L., 2011, "A Vision Inspection System for the Surface Defects of Strongly Reflected Metal Based on Multi-Class SVM," *Expert Syst. Appl.*, **38**(5), pp. 5930–5939.
- [11] Shanmugamani, R., Sadique, M., and Ramamoorthy, B., 2015, "Detection and Classification of Surface Defects of Gun Barrels Using Computer Vision and Machine Learning," *Measurement*, **60**, pp. 222–230.
- [12] Li, Q., Wang, M., and Gu, W., 2002, "Computer Vision Based System for Apple Surface Defect Detection," *Comput. Electron. Agric.*, **36**(2–3), pp. 215–223.
- [13] Pastor-López, I., Santos, I., Santamaría-Ibarika, A., Salazar, M., and Bringas, P. G., 2012, "Machine-Learning-Based Surface Defect Detection and Categorisation in High-Precision Foundry," 2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA), Singapore, July 18–20, IEEE, pp. 1359–1364.
- [14] Tao, X., Zhang, D., Ma, W., Liu, X., and Xu, D., 2018, "Automatic Metallic Surface Defect Detection and Recognition With Convolutional Neural Networks," *Appl. Sci.*, **8**(9), p. 1575.
- [15] Xie, X., 2008, "A Review of Recent Advances in Surface Defect Detection Using Texture Analysis Techniques," *ELCVIA: Electron. Lett. Comput. Vis. Image Anal.*, **7**(3), pp. 1–22.
- [16] Patel, B., and Bhaidasna, H., 2016, "Survey on Different Methods for Defect Detection," *Int. Res. J. Eng. Tech.*, **3**(2), pp. 1217–1220.
- [17] Hoang, D.-T., and Kang, H.-J., 2019, "A Survey on Deep Learning Based Bearing Fault Diagnosis," *Neurocomputing*, **335**, pp. 327–335.
- [18] Cao, W., Liu, Q., and He, Z., 2020, "Review of Pavement Defect Detection Methods," *IEEE Access*, **8**, pp. 14531–14544.
- [19] Luo, Q., Fang, X., Liu, L., Yang, C., and Sun, Y., 2020, "Automated Visual Defect Detection for Flat Steel Surface: A Survey," *IEEE Trans. Instrum. Meas.*, **69**(3), pp. 626–644.
- [20] Kumar, A., 2008, "Computer-Vision-Based Fabric Defect Detection: A Survey," *IEEE Trans. Ind. Electron.*, **55**(1), pp. 348–363.
- [21] Fouzia, M. T., and Nirmala, K., 2010, "A Literature Survey on Various Methods Used for Metal Defects Detection Using Image Segmentation," *Evaluation*, **5**(10), p. 8.
- [22] Czimmermann, T., Ciuti, G., Milazzo, M., Chiurazzi, M., Roccella, S., Oddo, C. M., and Dario, P., 2020, "Visual-Based Defect Detection and Classification Approaches for Industrial Applications—A Survey," *Sensors*, **20**(5), p. 1459.
- [23] Ahuja, S. K., and Shukla, M. K., 2017, "A Survey of Computer Vision Based Corrosion Detection Approaches," *International Conference on Information and Communication Technology for Intelligent Systems*, Ahmedabad, India, Mar. 25–26, Springer, pp. 55–63.
- [24] Goodfellow, I., Bengio, Y., and Courville, A., 2016, *Deep Learning*, MIT Press, Cambridge, MA.
- [25] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., 2014, "Generative Adversarial Nets," *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, NIPS'14*, Montréal, Canada, Dec. 8–13, MIT Press, pp. 2672–2680.
- [26] Kohonen, T., 1990, "The Self-organizing Map," *Proc. IEEE*, **78**(9), pp. 1464–1480.
- [27] He, K., Zhang, X., Ren, S., and Sun, J., 2016, "Deep Residual Learning for Image Recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, June 26–July 1, pp. 770–778.
- [28] Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K., 2016, "Squeezenet: Alexnet-Level Accuracy With 50x Fewer Parameters and <0.5 mb Model Size," *arXiv preprint arXiv:1602.07360*.
- [29] Yu, Z., Wu, X., and Gu, X., 2017, "Fully Convolutional Networks for Surface Defect Inspection in Industrial Environment," *International Conference on Computer Vision Systems*, Shenzhen, China, July 10–13, Springer, pp. 417–426.
- [30] Lin, Z., Guo, Z., and Yang, J., 2019, "Research on Texture Defect Detection Based on Faster-RCNN and Feature Fusion," *Proceedings of the 2019 11th International Conference on Machine Learning and Computing*, Zhuhai, China, Feb. 22–24, pp. 429–433.
- [31] "DAGM Data Set," <https://conferences.mpi-inf.mpg.de/dagm/2007/prizes.html>, Accessed January 2021.
- [32] "NEU Data Set," http://faculty.neu.edu.cn/yunhyan/NEU_surface_defect_database.html, Accessed January 2021.
- [33] "COCO Data Set," <https://cocodataset.org/#home>
- [34] He, Z., and Liu, Q., 2020, "Deep Regression Neural Network for Industrial Surface Defect Detection," *IEEE Access*, **8**, pp. 35 583–35 591.
- [35] Hochreiter, S., and Schmidhuber, J., 1997, "Long Short-Term Memory," *Neural Comput.*, **9**(8), pp. 1735–1780.
- [36] Li, Y., Huang, H., Xie, Q., Yao, L., and Chen, Q., 2018, "Research on a Surface Defect Detection Algorithm Based on Mobilenet-SSD," *Appl. Sci.*, **8**(9), p. 1678.
- [37] Simonyan, K., and Zisserman, A., 2015, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *International Conference on Learning Representations*, San Diego, CA, May 7–9.
- [38] Brinkmann, R., 2008, *The Art and Science of Digital Compositing: Techniques for Visual Effects, Animation and Motion Graphics*, Morgan Kaufmann, Burlington, MA.
- [39] Schaul, T., and Schmidhuber, J., 2010, "Metalearning," *Scholarpedia*, **5**(6), p. 4650.
- [40] Jolliffe, I., 1986, *Principal Component Analysis*, Springer Verlag, New York.
- [41] Chandola, V., Banerjee, A., and Kumar, V., 2009, "Anomaly Detection: A Survey," *ACM J.*, **41**(3), pp. 1–58.
- [42] Chalapathy, R., and Chawla, S., 2019, "Deep Learning for Anomaly Detection: A Survey," *CoRR*, <https://arxiv.org/abs/1901.03407>
- [43] Kiran, B., Thomas, D., and Parakkal, R., 2018, "An Overview of Deep Learning Based Methods for Unsupervised and Semi-Supervised Anomaly Detection in Videos," *J. Imagine*, **4**(2), p. 36.
- [44] Wulsin, D., Blanco, J., Mani, R., and Litt, B., 2010, "Semi-Supervised Anomaly Detection for Eeg Waveforms Using Deep Belief Nets," 2010 Ninth

- International Conference on Machine Learning and Applications, Washington, DC, Dec. 12–14, pp. 436–441.
- [45] Song, H., Jiang, Z., Men, A., and Yang, B., 2017, “A Hybrid Semi-Supervised Anomaly Detection Model for High-Dimensional Data,” *J. Comput. Intell. Neurosci.*, **2017**.
- [46] Ruff, L., Vandermeulen, R. A., Görnitz, N., Binder, A., Müller, E., Müller, K.-R., and Kloft, M., 2019, “Deep Semi-Supervised Anomaly Detection,” arXiv preprint arXiv:1906.02694.
- [47] Schlegl, T., Seeßböck, P., Waldstein, S. M., Schmidt-Erfurth, U., and Langs, G., 2017, “Unsupervised Anomaly Detection With Generative Adversarial Networks to Guide Marker Discovery,” *Information Processing in Medical Imaging*, Niethammer, M., Styner, M., Aylward, S., Zhu, H., Oguz, I., Yap, P.-T., and Shen, D., eds., Springer International Publishing, New York, pp. 146–157.
- [48] Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., Müller, E., and Kloft, M., 2018, “Deep One-Class Classification,” *Proceedings of the 35th International Conference on Machine Learning*, YDyap, J., and Krause, A., eds., Vol. 80, Proceedings of Machine Learning Research, PMLR, pp. 4393–4402.
- [49] Xu, H., and Huang, Z., 2018, “Research on Target Detection Methods Under the Concept of Deep Learning,” *J. Phys.: Conference Ser.*, **1087**(6), p. 062055.
- [50] Wang, X., and Hu, Z., 2017, “Grid-Based Pavement Crack Analysis Using Deep Learning,” 2017 4th International Conference on Transportation Information and Safety (ICTIS), Hong Kong, July 1–3, pp. 917–924.
- [51] Xie, L., Xiang, X., Xu, H., Wang, L., Lin, L., and Yin, G., 2020, “Ffcnn: A Deep Neural Network for Surface Defect Detection of Magnetic Tile,” *IEEE Trans. Ind. Electron.*, **68**(4), p. 1.
- [52] Maestro-Watson, D., Balzategui, J., Eciolaza, L., and Arana-Arextola, N., 2018, “Deep Learning for Deflectometric Inspection of Specular Surfaces,” The 13th International Conference on Soft Computing Models in Industrial and Environmental Applications, San Sebastian, Spain, June 6–8, Springer, pp. 280–289.
- [53] Fu, G., Sun, P., Zhu, W., Yang, J., Cao, Y., Yang, M. Y., and Cao, Y., 2019, “A Deep-Learning-Based Approach for Fast and Robust Steel Surface Defects Classification,” *Optics Lasers Eng.*, **121**, pp. 397–405.
- [54] Racki, D., Tomaževič, D., and Skočaj, D., “Towards Surface Anomaly Detection With Deep Learning,” International Electrotechnical and Computer Science Conference (ERK), Portorož, Slovenia, Sept. 25–26, pp. 437–440.
- [55] Wu, X., Cao, K., and Guo, X., 2017, “A Surface Defect Detection Based on Convolutional Neural Network,” International Conference on Computer Vision Systems, Shenzhen, China, July 10–13, Springer, pp. 185–194.
- [56] Soukup, D., and Huber-Mörk, R., 2014, “Convolutional Neural Networks for Steel Surface Defect Detection From Photometric Stereo Images,” International Symposium on Visual Computing, Las Vegas, NV, Dec. 8–10, Springer, pp. 668–677.
- [57] Azizah, L. M., Umayah, S. F., Riyadi, S., Damarjati, C., and Utama, N. A., 2017, “Deep Learning Implementation Using Convolutional Neural Network in Mangosteen Surface Defect Detection,” 2017 7th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, Nov. 24–26, IEEE, pp. 242–246.
- [58] Cha, Y.-J., Choi, W., and Büyükoztürk, O., 2017, “Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks,” *Comput. Aided Civil Infrastructure Eng.*, **32**(5), pp. 361–378.
- [59] Ren, R., Hung, T., and Tan, K. C., 2017, “A Generic Deep-Learning-Based Approach for Automated Surface Inspection,” *IEEE Trans. Cybern.*, **48**(3), pp. 929–940.
- [60] Kang, G., Gao, S., Yu, L., and Zhang, D., 2018, “Deep Architecture for High-Speed Railway Insulator Surface Defect Detection: Denoising Autoencoder With Multitask Learning,” *IEEE Trans. Instrum. Meas.*, **68**(8), pp. 2679–2690.
- [61] Lien, P. C., and Zhao, Q., 2018, “Product Surface Defect Detection Based on Deep Learning,” 2018 IEEE 16th International Conference on Dependable, Autonomic and Secure Computing, 16th International Conference on Pervasive Intelligence and Computing, 4th International Conference on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DAASC/PICOM/DataCom/CyberSciTech), Athens, Greece, Aug. 12–15, IEEE, pp. 250–255.
- [62] Tout, K., Bouabdellah, M., Cudel, C., and Urban, J.-P., 2019, “Automated Vision System for Crankshaft Inspection Using Deep Learning Approaches,” Fourteenth International Conference on Quality Control by Artificial Vision, Alsace, France, May 15–17, Vol. 11172, International Society for Optics and Photonics, p. 111720N.
- [63] Song, L., Li, X., Yang, Y., Zhu, X., Guo, Q., and Yang, H., 2018, “Detection of Micro-Defects on Metal Screw Surfaces Based on Deep Convolutional Neural Networks,” *Sensors*, **18**(11), p. 3709.
- [64] Xu, X., Zheng, H., Guo, Z., Wu, X., and Zheng, Z., 2019, “Sdd-cnn: Small Data-Driven Convolutional Neural Networks for Subtle Roller Defect Inspection,” *Appl. Sci.*, **9**(7), p. 1364.
- [65] Sun, X., Gu, J., Huang, R., Zou, R., and Giron Palomares, B., 2019, “Surface Defects Recognition of Wheel Hub Based on Improved Faster R-CNN,” *Electronics*, **8**(5), p. 481.
- [66] Wang, T., Chen, Y., Qiao, M., and Snoussi, H., 2018, “A Fast and Robust Convolutional Neural Network-Based Defect Detection Model in Product Quality Control,” *Int. J. Adv. Manuf. Technol.*, **94**(9–12), pp. 3465–3471.
- [67] Qiu, L., Wu, X., and Yu, Z., 2019, “A High-Efficiency Fully Convolutional Networks for Pixel-Wise Surface Defect Detection,” *IEEE Access*, **7**, pp. 15 884–15 893.
- [68] Lai, Y. K., and Hu, J., 2018, “A Texture Generation Approach for Detection of Novel Surface Defects,” 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, Oct. 7–10, pp. 4357–4362.
- [69] Sison, H., Konghuayrob, P., and Kaitwanidvilai, S., 2018, “A Convolutional Neural Network for Segmentation of Background Texture and Defect on Copper Clad Lamination Surface,” 2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST), Phuket, Thailand, July 4–7, pp. 1–4.
- [70] Faghih-Roohi, S., Hajizadeh, S., Núñez, A., Babuska, R., and De Schutter, B., 2016, “Deep Convolutional Neural Networks for Detection of Rail Surface Defects,” 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, July 24–29, pp. 2584–2589.
- [71] Lian, J., Jia, W., Zareapoor, M., Zheng, Y., Luo, R., Jain, D. K., and Kumar, N., 2020, “Deep-Learning-Based Small Surface Defect Detection Via an Exaggerated Local Variation-Based Generative Adversarial Network,” *IEEE Trans. Ind. Inform.*, **16**(2), pp. 1343–1351.
- [72] Li, K., Wang, X., and Ji, L., 2019, “Application of Multi-Scale Feature Fusion and Deep Learning in Detection of Steel Strip Surface Defect,” 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM), Dublin, Ireland, Oct. 17–19, pp. 656–661.
- [73] Xu, L., Lv, S., Deng, Y., and Li, X., 2020, “A Weakly Supervised Surface Defect Detection Based on Convolutional Neural Network,” *IEEE Access*, **8**, pp. 42 285–42 296.
- [74] Khumaidi, A., Yuniarno, E. M., and Purnomo, M. H., 2017, “Welding Defect Classification Based on Convolution Neural Network (CNN) and Gaussian Kernel,” 2017 International Seminar on Intelligent Technology and Its Applications (ISITIA), Surabaya, Indonesia, Aug. 28–29, pp. 261–265.
- [75] Park, J.-K., Kwon, B.-K., Park, J.-H., and Kang, D.-J., 2016, “Machine Learning-Based Imaging System for Surface Defect Inspection,” *Int. J. Precision Eng. Manuf. Green Technol.*, **3**(3), pp. 303–310.
- [76] Deng, Z., Yan, X., Zhang, S., and Bailey, C. P., 2020, “Extremal Region Analysis Based Deep Learning Framework for Detecting Defects,” arXiv preprint arXiv:2003.08525.
- [77] Yuan, Z.-C., Zhang, Z.-T., Su, H., Zhang, L., Shen, F., and Zhang, F., 2018, “Vision-Based Defect Detection for Mobile Phone Cover Glass Using Deep Neural Networks,” *Int. J. Precision Eng. Manuf. Green Technol.*, **19**(6), pp. 801–810.
- [78] Wei, R., and Bi, Y., 2019, “Research on Recognition Technology of Aluminum Profile Surface Defects Based on Deep Learning,” *Materials*, **12**(10), p. 1681.
- [79] Staar, B., Lütjen, M., and Freitag, M., 2019, “Anomaly Detection With Convolutional Neural Networks for Industrial Surface Inspection,” *Procedia CIRP*, **79**(1), pp. 484–489.
- [80] Li, Y., Zhao, W., and Pan, J., 2016, “Deformable Patterned Fabric Defect Detection With Fisher Criterion-Based Deep Learning,” *IEEE Trans. Autom. Sci. Eng.*, **14**(2), pp. 1256–1264.
- [81] Natarajan, V., Hung, T., Vaikundam, S., and Chia, L., 2017, “Convolutional Networks for Voting-Based Anomaly Classification in Metal Surface Inspection,” 2017 IEEE International Conference on Industrial Technology (ICIT), Singapore, Dec. 27–29, pp. 986–991.
- [82] Zhang, B., Hong, K.-M., and Shin, Y. C., 2020, “Deep-Learning-Based Porosity Monitoring of Laser Welding Process,” *Manuf. Lett.*, **23**, pp. 62–66.
- [83] Mujeeb, A., Dai, W., Erdt, M., and Sourin, A., 2018, “Unsupervised Surface Defect Detection Using Deep Autoencoders and Data Augmentation,” 2018 International Conference on Cyberworlds (CW), Caen, France, Sept. 28–30, IEEE, pp. 391–398.
- [84] Volkau, I., Abdul, M., Dai, W., Erdt, M., and Sourin, A., 2019, “Detection Defect in Printed Circuit Boards Using Unsupervised Feature Extraction Upon Transfer Learning,” 2019 International Conference on Cyberworlds (CW), Kyoto, Japan, Oct. 2–4, IEEE, pp. 101–108.
- [85] Mei, S., Yang, H., and Yin, Z., 2018, “An Unsupervised-Learning-Based Approach for Automated Defect Inspection on Textured Surfaces,” *IEEE Trans. Instrum. Meas.*, **67**(6), pp. 1266–1277.
- [86] Zhang, A., Wang, K. C., Li, B., Yang, E., Dai, X., Peng, Y., Fei, Y., Liu, Y., Li, J. Q., and Chen, C., 2017, “Automated Pixel-Level Pavement Crack Detection on 3d Asphalt Surfaces Using a Deep-Learning Network,” *Comput. Aided Civil Infrastructure Eng.*, **32**(10), pp. 805–819.
- [87] Masci, J., Meier, U., Ciresan, D., Schmidhuber, J., and Fricout, G., 2012, “Steel Defect Classification With Max-Pooling Convolutional Neural Networks,” The 2012 International Joint Conference on Neural Networks (IJCNN), Brisbane, Australia, June 10–15, IEEE, pp. 1–6.
- [88] Racki, D., Tomaževič, D., and Skočaj, D., 2018, “The Effect of Different CNN Configurations on Textured-Surface Defect Segmentation and Detection Performance,” 23rd Computer Vision Winter Workshop, Český Krumlov, Czech Republic, Feb. 5–7.
- [89] Liong, S.-T., Gan, Y., Huang, Y.-C., Yuan, C.-A., and Chang, H.-C., 2019, “Automatic Defect Segmentation on Leather With Deep Learning,” arXiv preprint arXiv:1903.12139.
- [90] Akhyar, F., Lin, C.-Y., Muchtar, K., Wu, T.-Y., and Ng, H.-F., 2019, “High Efficient Single-Stage Steel Surface Defect Detection,” 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Taipei, Taiwan, Sept. 18–21, IEEE, pp. 1–4.
- [91] Akhyar, F., Hsu, C.-Y., and Ng, H.-F., 2019, “Cascading Convolutional Neural Network for Steel Surface Defect Detection,” Advances in Artificial Intelligence, Software and Systems Engineering: Proceedings of the AHFE 2019 International Conference on Human Factors in Artificial Intelligence and Social Computing, the AHFE International Conference on Human Factors, Software, Service and Systems Engineering, and the AHFE International

- Conference of Human Factors in Energy, Vol. 965, Washington DC, July 24–28, Springer, pp. 202.
- [92] Li, X., Zhou, Y., and Chen, H., 2020, “Rail Surface Defect Detection Based on Deep Learning,” Eleventh International Conference on Graphics and Image Processing (ICGIP 2019), Hangzhou, China, Oct. 12–14, Vol. 11373, International Society for Optics and Photonics, p. 113730K.
- [93] Cheon, S., Lee, H., Kim, C. O., and Lee, S. H., 2019, “Convolutional Neural Network for Wafer Surface Defect Classification and the Detection of Unknown Defect Class,” *IEEE Trans. Semiconductor Manuf.*, **32**(2), pp. 163–170.
- [94] Yuan, H., Chen, H., Liu, S., Lin, J., and Luo, X., 2019, “A Deep Convolutional Neural Network for Detection of Rail Surface Defect,” 2019 IEEE Vehicle Power and Propulsion Conference (VPPC), Hanoi, Vietnam, Oct. 14–17, IEEE, pp. 1–4.
- [95] He, Y., Song, K., Meng, Q., and Yan, Y., 2019, “An End-to-End Steel Surface Defect Detection Approach Via Fusing Multiple Hierarchical Features,” *IEEE Trans. Instrum. Meas.*, **69**(4), pp. 1493–1504.
- [96] Dong, H., Song, K., He, Y., Xu, J., Yan, Y., and Meng, Q., 2019, “Pga-net: Pyramid Feature Fusion and Global Context Attention Network for Automated Surface Defect Detection,” *IEEE Trans. Ind. Inform.*, **16**(12), pp. 7448–7458.
- [97] Gu, M., Huang, D., Zhou, X., Li, Y., and Li, Y., 2020, “Research on Intelligent Detection Technology of Surface Defects of Nuclear Fuel Rods Based on Machine Vision,” Proceedings of the Seventh Asia International Symposium on Mechatronics, Springer, pp. 927–936.
- [98] Mujeeb, A., Dai, W., Erdt, M., and Sourin, A., 2019, “One Class Based Feature Learning Approach for Defect Detection Using Deep Autoencoders,” *Adv. Eng. Inform.*, **42**, p. 100933.
- [99] Di, H., Ke, X., Peng, Z., and Dongdong, Z., 2019, “Surface Defect Classification of Steels With a New Semi-Supervised Learning Method,” *Optics Lasers Eng.*, **117**, pp. 40–48.
- [100] Li, Y. T., and Guo, J. I. G., 2018, “A VGG-16 Based Faster RCNN Model for PCB Error Inspection in Industrial AOI Applications,” 2018 International Conference on Consumer Electronics-Taiwan(ICCE-TW), Taiwan, China, May 19–21, IEEE.
- [101] Baumgartl, H., Tomas, J., Buettner, R., and Merkel, M., 2020, “A Deep Learning-Based Model for Defect Detection in Laser-Powder Bed Fusion Using In-Situ Thermographic Monitoring,” *Progress in Additive Manuf.*, **2020**(5), pp. 1–9.
- [102] Qu, Z., Shen, J., Li, R., Liu, J., and Guan, Q., 2018, “Partsnet: A Unified Deep Network for Automotive Engine Precision Parts Defect Detection,” Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence, Stockholm, Sweden, July 23–25, pp. 594–599.
- [103] Chen, J., Liu, Z., Wang, H., Núñez, A., and Han, Z., 2017, “Automatic Defect Detection of Fasteners on the Caterpillar Support Device Using Deep Convolutional Neural Network,” *IEEE Trans. Instrum. Meas.*, **67**(2), pp. 257–269.
- [104] Tabernik, D., Šela, S., Skvarč, J., and Škočaj, D., 2019, “Segmentation-Based Deep-Learning Approach for Surface-Defect Detection,” *J. Intell. Manuf.*, **31**(3), pp. 1–18.
- [105] Chen, H., Pang, Y., Hu, Q., and Liu, K., 2018, “Solar Cell Surface Defect Inspection Based on Multispectral Convolutional Neural Network,” *J. Intell. Manuf.*, **31**(2), pp. 1–16.
- [106] Weimer, D., Scholz-Reiter, B., and Shpitalni, M., 2016, “Design of Deep Convolutional Neural Network Architectures for Automated Feature Extraction in Industrial Inspection,” *CIRP Ann.*, **65**(1), pp. 417–420.
- [107] Shang, L., Yang, Q., Wang, J., Li, S., and Lei, W., 2018, “Detection of Rail Surface Defects Based on CNN Image Recognition and Classification,” 2018 20th International Conference on Advanced Communication Technology (ICACT), Chennai, India, Feb. 2–3, IEEE, pp. 45–51.
- [108] Konrad, T., Lohmann, L., and Abell, D., 2019, “Surface Defect Detection for Automated Inspection Systems Using Convolutional Neural Networks,” 2019 27th Mediterranean Conference on Control and Automation (MED), Akko, Israel, July 1–4, IEEE, pp. 75–80.
- [109] Liu, Y., Xu, K., and Xu, J., 2019, “Periodic Surface Defect Detection in Steel Plates Based on Deep Learning,” *Appl. Sci.*, **9**(15), p. 3127.
- [110] Xiao, L., Wu, B., and Hu, Y., 2020, “Surface Defect Detection Using Image Pyramid,” *IEEE Sens. J.*, **20**(13), pp. 7181–7188.
- [111] Xiao, L., Lu, M., and Huang, H., 2020, “Detection of Powder Bed Defects in Selective Laser Sintering Using Convolutional Neural Network,” *Int. J. Adv. Manuf. Technol.*, pp. 1–12.
- [112] Cui, W., Zhang, Y., Zhang, X., Li, L., and Liou, F., 2020, “Metal Additive Manufacturing Parts Inspection Using Convolutional Neural Network,” *Appl. Sci.*, **10**(2), p. 545.
- [113] Hartl, R., Landgraf, J., Spahl, J., Bachmann, A., and Zaeh, M. F., 2019, “Automated Visual Inspection of Friction Stir Welds: A Deep Learning Approach,” Multimodal Sensing: Technologies and Applications, Munich, Germany, June 26–27, Vol. 11059, International Society for Optics and Photonics, p. 1105909.
- [114] Li, Y., Li, H., and Wang, H., 2018, “Pixel-Wise Crack Detection Using Deep Local Pattern Predictor for Robot Application,” *Sensors*, **18**(9), p. 3042.
- [115] Sun, X., Gu, J., Tang, S., and Li, J., 2018, “Research Progress of Visual Inspection Technology of Steel Products-A Review,” *Appl. Sci.*, **8**(11), p. 2195.
- [116] Kholief, E. A., Darwish, S. H., and Fors, N., 2017, “Detection of Steel Surface Defect Based on Machine Learning Using Deep Auto-Encoder Network,” *Ind. Eng. Oper. Manage.*, pp. 218–229.
- [117] Shi, Y., Cui, L., Qi, Z., Meng, F., and Chen, Z., 2016, “Automatic Road Crack Detection Using Random Structured Forests,” *IEEE Trans. Intell. Transp. Syst.*, **17**(12), pp. 3434–3445.
- [118] Gan, J., Li, Q., Wang, J., and Yu, H., 2017, “A Hierarchical Extractor-Based Visual Rail Surface Inspection System,” *IEEE Sens. J.*, **17**(23), pp. 7935–7944.
- [119] Silvestre-Blanes, J., Albero-Albero, T., Miralles, I., Pérez-Llorens, R., and Moreno, J., 2019, “A Public Fabric Database for Defect Detection Methods and Results,” *Autex Res. J.*, **19**(4), pp. 363–374.
- [120] Song, K., and Yan, Y., 2013, “Micro Surface Defect Detection Method for Silicon Steel Strip Based on Saliency Convex Active Contour Model,” *Math. Probl. Eng.*, **2013**.
- [121] Girshick, R., Donahue, J., Darrell, T., and Malik, J., 2014, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, June 23–28, pp. 580–587.
- [122] Girshick, R., 2015, “Fast R-CNN,” Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, Dec. 7–13, pp. 1440–1448.
- [123] Ren, S., He, K., Girshick, R., and Sun, J., 2015, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” Advances in Neural Information Processing Systems, Montreal, Canada, Dec. 7–10, pp. 91–99.
- [124] Shorten, C., and Khoshgoftaar, T., 2019, “A Survey on Image Data Augmentation for Deep Learning,” *J. Big Data*, **6**(1), p. 1106.
- [125] Xie, N., Ras, G., van Gerven, M., and Doran, D., 2020, “Explainable Deep Learning: A Field Guide for the Uninitiated,” arXiv preprint arXiv:2004.14545.
- [126] Poggio, T., Kawaguchi, K., Liao, Q., Miranda, B., Rosasco, L., Boix, X., Hidary, J., and Mhaskar, H., 2017, “Theory of Deep Learning III: Explaining the Non-Overfitting Puzzle,” arXiv preprint arXiv:1801.00173.
- [127] Pan, S. J., and Yang, Q., 2009, “A Survey on Transfer Learning,” *IEEE Trans. Knowl. Data Eng.*, **22**(10), pp. 1345–1359.
- [128] Samet, R., Bayram, A., Tural, S., and Aydin, S., 2016, “Primer Defects Detection on Military Cartridge Cases,” 2016 Nicograph International (NicoInt) 2016, Hanzhou, China, July 6, IEEE, pp. 96–99.
- [129] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S., 2017, “Feature Pyramid Networks for Object Detection,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, July 21–26, pp. 2117–2125.
- [130] Hao, Z., Liu, Y., Qin, H., Yan, J., Li, X., and Hu, X., 2017, “Scale-Aware Face Detection,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, July 21–26, pp. 6186–6195.
- [131] Welleck, S., Mao, J., Cho, K., and Zhang, Z., 2017, “Saliency-Based Sequential Image Attention With Multiset Prediction,” Advances in Neural Information Processing Systems, Long Beach, CA, Dec. 4–9, pp. 5173–5183.
- [132] Fang, Y., Kuan, K., Lin, J., Tan, C., and Chandrasekhar, V., 2017, “Object Detection Meets Knowledge Graphs,” Proceedings of the 26th International Joint Conference on Artificial Intelligence, IJCAI’17, Melbourne, Australia, Aug. 19–25, AAAI Press, pp. 1661–1667.
- [133] Gavrichchaka, V., Senyukova, O., and Koepke, M., 2019, “Synergy of Physics-Based Reasoning and Machine Learning in Biomedical Applications: Towards Unlimited Deep Learning With Limited Data,” *Adv. Phys.: X*, **4**(1), p. 1582361.