Unsupervised Defect Detection for Infrastructure Inspection

N. P. García-de-la-Puente, Rocío del Amor, Fernando García-Torres, Adrián Colomer, and Valery Naranjo

Instituto Universitario de Investigación en Tecnología Centrada en el Ser Humano, *Universitat Politècnica de València*, Valencia, Spain
vnaranjo@dcom.upv.es

Abstract. Artificial Intelligence (AI) provides a fundamental aid in building operations, allowing infrastructure inspection and compliance with safety standards. In the collaborative tasks involved, detecting areas of interest, such as surface defects, is crucial. A drawback of supervised AI-based approaches is that they require manual annotation, which entails additional costs. This paper presents a novel unsupervised anomaly detection approach for locating defects based on generative models that learn the distribution of defect-free images. Using attention maps to validate in a subset, we propose a formulation that does not require accessing labelled images, enabling task automation, maintenance optimisation and cost reduction.

Keywords: Visual Inspection · Infrastructure Inspection · Defects · Unsupervised Segmentation.

1 Introduction

Collaborative Networks (CN) are alliances of entities working together to solve complex problems that a single individual or organisation cannot efficiently tackle. Three factors have led Artificial Intelligence (AI) to show promising results in favour of CNs. Firstly, the increase in data availability and lower procurement hardware prices. Secondly, the development of techniques that allow processing large datasets to identify patterns, correlations and anomalies. Finally, the integration and interoperability, bridging the gap between systems and sharing knowledge. When combined, CN and AI can solve complex problems requiring human expertise and the automation of algorithms.

In the world of AI, one field of great interest is Computer Vision (CV). It is a huge field focused on automatically extracting information from visual content, such as images or videos. The implementation of CV has significantly increased because of the integration of AI and Deep learning (DL) techniques in the areas of biomedicine, the food industry, the agricultural sector and the development of autonomous cars, among others [1]–[3]. One of the most widespread applications is related to product inspection in the industry, either during the production phase of the parts or in the operation of products and structures [4].

Various algorithms have been used in CV for industry manufacturing and component performance monitoring: supervised, semi-supervised and unsupervised [5]. Supervised algorithms learn to recognise patterns in images, which allows to predict the presence of defects in unseen images. However, supervised learning requires large amounts of data and duly labelled by experts, which entails considerable time and economic costs. In contrast, unsupervised learning is used when the data is unlabelled. The goal is to find hidden patterns or structures in the data without prior knowledge. Unsupervised anomaly detection can be employed when labelled data is unavailable. This approach is based on a few assumptions, including that most samples in the dataset are expected not defective and defective instances are rare occurrences within the dataset [6]. Semi-supervised algorithms combine features of supervised and unsupervised algorithms. While unsupervised learning has shown promise in some CV applications, it is less commonly used for defect inspection than supervised learning, given that the latter has traditionally performed better.

Inspecting parts and detecting defects in the industry are generally conducted using Non-Destructive Testing (NDT). A diverse range of quality assessment and defect inspection methods are available. Their selection depends on the specific product being examined and the significance of the information obtained from each technique in a particular scenario. The most commonly used methods are visual inspection, thermography, acoustic emissions, electromagnetic testing, ultrasound or X-rays. CV has been successfully integrated into NDT, highlighting its use with thermal imaging cameras to detect heat anomalies that may indicate defects [7] or with X-ray imaging to detect internal flaws in components [8]. Combining CV technology with other NDT methods can improve defect detection capabilities, reduce the risk of product failure and increase overall product quality.

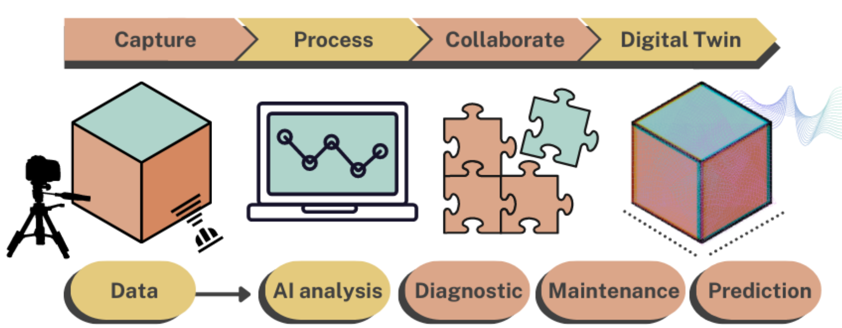

Fig. 1. Digital Twin flow as a non-destructive test for defect inspection.

Digital Twin (DT) technology is also being explored as a NDT for defect inspection. Creating a virtual replica of a product or component can simulate various operating conditions and analyse the impact of defects on product performance without physical testing. This approach can significantly reduce the time and cost associated with traditional testing methods while providing more accurate and detailed data. Leveraging DT as a NDT for defect inspection can enhance product quality, minimise the risk of failure, and increase efficiency in the manufacturing process [9]. AI can be employed to analyse the data obtained from DT (Fig. 1), facilitating diagnostics and preventive maintenance. Techniques such as explainable AI can be incorporated for data analysis and to predict potential risks [10].

The present research work is part of a more comprehensive development project intended to add value to the construction and operation of the building. Specifically, the aim is to develop a model for inspecting infrastructures to ensure compliance with safety standards. As a novelty, unsupervised techniques are to be used with an end-to-end approach to perform semantic segmentation and classification at the same time. This allows circumventing the usual lack of labelled data and provide a method that can achieve a high level of accuracy on unlabelled data, thus avoiding the costs and time associated with supervised tasks. To validate the accuracy and performance of the model proposed, a labelled subset of the data is intended to be used.

2 Related work: Defect detection in civil infraestructures

Defect detection with AI-based CV models has the potential to revolutionise the way that civil infrastructure and construction are monitored and maintained. DL is becoming increasingly popular for detecting and assessing civil infrastructure and construction defects. It has been the focus of research studies by many authors [11], [12]. Several DL algorithms have been used or developed for the classification, localisation, or segmentation of defects, mainly consisting of cracks in concrete or steel structures and, to a lesser extent, corrosion [13].

The most used algorithms are supervised convolutional neural network (CNN) variations with end-to-end DL approaches. These algorithms have reached great results for defect detection in concrete structures. For example, a CNN was employed in a study to construct a model that can effectively identify concrete cracks in various scenarios, overcoming the restrictions posed by conventional image-processing techniques [14]. Yang et al. introduced a dual CNN architecture composed of a CNN and a fully connected network (FCN) for detecting cracks on concrete bridges. The CNN method was utilised to eliminate interference signals, while the FCN was employed to extract relevant features of the cracks [15]. Cha et al. used CNNs to detect and classify cracks, reaching good results in detection accuracy [16]. Nevertheless, these supervised approaches have the crucial barrier of the absence of publicly available datasets with appropriate annotations due to the necessity for time-consuming and labour-intensive work for data preparation [17].

Regarding the segmentation of cracks using CV, some authors studied the localisation and delineation of the damaged shape using deep CNN. Islam et al. developed a transfer-learning approach to identify cracks using four pretrained models[18]. Zhang et al. used a pixel-wise CNN and an FCN to segment defects such as concrete cracks, concrete spalling, exposed reinforcement bars, steel corrosion, steel fracture and fatigue cracks, and asphalt [19].

Unsupervised machine learning methods have been used for crack detection in construction in the last years, such as Principal Component Analysis (PCA) and several clustering methods, but they need more robustness. As far as we are concerned, there need to be more approaches for training DL algorithms without annotation that have been used for defect detection in other industrial fields but to a lesser degree in construction and civil infrastructure [20]. Although some studies use DL algorithms, there are yet to be state-of-the-art studies that perform the classification and segmentation of cracks [21], [22]. Thus, the application of unsupervised learning in this field is still relatively new and more research is needed to explore its full potential.

3 Methodology

An overview of our proposed method for defect detection is depicted in Figure 2. In the following, we describe the problem formulation and each proposed component.

Problem formulation: Under the paradigm of unsupervised anomaly detection, we denote the set of unlabeled training images as $X_T = \{x_n\}_{n=1}^{N}$, composed of only normal images, i.e., images without defects. We now define an encoder in charge of transforming the input data $\mathbf{X}_T$ into a latent representation (with a lower dimensionality) $\mathbf{Z}$ through a non-linear mapping function, $\mathbf{Z} = f_\phi(\mathbf{X}_T)$, where ϕ are the learnable parameters of the encoder architecture. The decoder stage produces the reconstruction of the data based on the features embedded in the latent space, $\mathbf{R} = g_\theta(\mathbf{Z})$. The reconstructed representation $\mathbf{R}$ is required to be as similar to $\mathbf{X}_T$ as possible. During the inference, we use a set of unlabeled images $X_I = \{x_m\}_{m=1}^{M}$, composed of images without and with defects to differentiate them and provide the map where the defect is found.

3.1 Variational autoencoder

Variational autoencoder (VAE) is an unsupervised approach composed of an encoder-decoder architecture commonly used for anomaly detection [23]. With a VAE, the input data is coded as a multivariate normal distribution $p(z|x)$ around a point in the latent space. In this way, the encoder part is optimized to obtain a multivariate normal distribution's mean and covariance matrix. The VAE algorithm assumes no correlation exists between latent space dimensions; therefore, the covariance matrix is diagonal. In this way, the encoder only needs to assign each input sample to a mean and a variance vector. In addition, the logarithm of the variance is set, as this can take any real number in the range

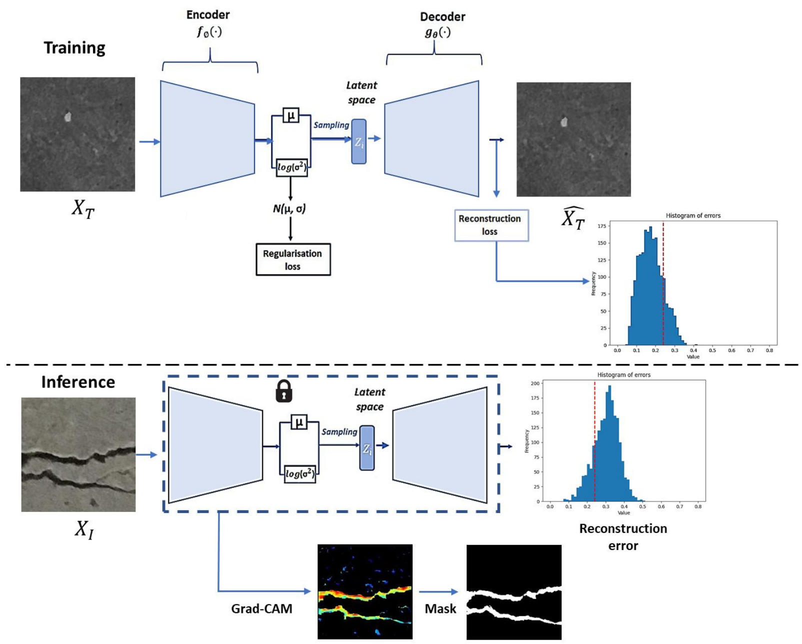

Fig. 2. Method overview. In line with the established literature on anomaly detection, the Variational Autoencoder (VAE) is optimized to maximize the evidence lower bound (ELBO). Furthermore, we incorporate an attention constraint through a size-constrained loss, which compels the network to explore the entire image. During inference, the attention map is thresholded to generate the final segmentation mask.

$(-\infty, \infty)$, matching the natural output range from a neural network, whereas that variance values are always positive. To provide continuity and completeness to the latent space, it is necessary to regularize both the logarithm of the variance and the mean of the distributions returned by the encoder. This regularisation is achieved by matching the encoder output distribution to the standard normal distribution ($z_\mu = 0$ and $z_\sigma = 1$). After obtaining and optimizing the parameters of the mean and variance of the latent distributions, it is necessary to take samples of the learned representations to reconstruct the original input data. Samples of the encoder output distribution are obtained as follows:

$$Z \approx p(z|x) = z_\mu + z_\sigma \cdot \epsilon \tag{1}$$

where ϵ is randomly sampled from a standard normal distribution and $\sigma = \exp(\frac{\log(\sigma^2)}{2})$.

The minimized loss function in a variational autoencoder comprises two terms: (1) a reconstruction term that compares the reconstructed data to the original input to get as effective encoding-decoding as possible and (2) a regularisation term in charge of regularizing the latent space organization. The regularisation term is expressed as the *Kulback-Leibler* (KL) divergence that measures the difference between the predicted latent probability distribution of the data and the standard normal distribution [24]:

$$D_{KL}[N(z_\mu, z_\sigma)||N(0,1)] = \frac{1}{2}\sum(1 + log(z_\sigma^2) - z_\mu^2 - z_\sigma^2) \quad (2)$$

The KL function is minimised to 0 if $\mu = 0$ and $log(\sigma^2) = 0$ for all dimensions. As these two terms differ from 0, the variational autoencoder loss increases. The compensation between the reconstruction error and the KL divergence is a hyper-parameter to be adjusted in this type of architecture.

Since training a VAE consists in minimizing a two-term loss function, this is equivalent to maximize the evidence lower-bound (ELBO):

$$\mathcal{L}_{VAE} = \mathcal{L}_R(x_T, \hat{x}_T) + \beta\mathcal{L}_{KL}(p(z|x)||p(z)) \quad (3)$$

where β is a weighting factor to optimize.

3.2 Anomaly segmentation via Grad-CAMs

Several works based on unsupervised anomaly detection use attention maps to mimic the segmentation mask of the anomaly. In particular, attention maps $\mathbf{a} \in \mathbb{R}^{\Omega_i}$ are generated from the mean latent vector z_μ, by using Grad-CAM [25] via backpropagation to the encoder block output. Therefore, given an image ($\mathbf{x}$), the attention maps is calculated as follows:

$$a = \sigma\left(\sum_k^K \alpha_k f_\phi(x)_k\right) \quad (4)$$

where K is the total number of filters of the encoder layer, σ is the sigmoid function, and α_k is the generated gradient such that: $\alpha_k = \frac{1}{|a|}\sum_{t\in\Omega_T}\frac{\partial z_\mu}{\partial a_{k,t}}$, where Ω_T is the spatial feature domain.

The idea underlying the Grad-CAM is to enforce them to cover the whole free-defect image without showing high activations concentrated in some areas. In inference, the activations will be concentrated in the area with defects. Therefore, it is necessary to introduce a constraint related to the Grad-CAM in the global loss function. Following the method show in [26], we use a log-barrier extension function with a single global constraint to achieve maximum coverage of class-activation maps over the whole image. Thus, we can formally define the approximation of log-barrier as:

$$\psi_t(z) = \begin{cases} -\frac{1}{t}log(-f_c(a)) & \text{if } f_c(a) \leq -\frac{1}{t^2} \\ tf_c(a) - \frac{1}{t}log(\frac{1}{t^2}) + \frac{1}{t} & \text{otherwise,} \end{cases} \quad (5)$$

where t controls the barrier during training, and $f_c(a) = \left(1 - \frac{1}{|\Omega_T|}\sum_{l\in\Omega} a_l\right)$ is the constraint over the attention map from the jth image, which enforces the generated attention map to cover the whole image, where Ω is the spatial features domain.

$$\mathcal{L} = \mathcal{L}_{VAE} + \lambda \sum_{n=1}^{N} \psi_t (1 - \frac{1}{|\Omega_T|} \sum_{l \in \Omega_i} a_l) \tag{6}$$

In this scenario, for a given t, the optimizer will try to find a solution with a good compromise between minimizing the loss of the VAE and satisfying the constraint $f_c(a)$.

3.3 Inference

During inference, we obtain the reconstruction of the inferred images X_I using the VAE trained with defect-free images (X_T). After obtaining the reconstructed images, we calculate the error concerning the original images. According to our hypothesis, we consider that those inferred images with a high reconstruction error will be defective. Therefore, we establish a threshold by considering the mean and standard deviation of the errors in the defect-free images used for validation. Inferred images with an error exceeding the threshold will be classified as defective, while those falling below the threshold will be classified as non-defective. Additionally, we use the anomaly saliency map as a segmentation map. During the experimental stage, we found that anomalies produce larger activation on attention maps than the constrained normal samples. Then, the map is thresholded to create an anomaly mask of the image.

4 Experimental setting

4.1 Dataset and evaluation metrics

The dataset utilized in this study [27] comprises concrete images which were sourced from multiple buildings within the METU Campus. The dataset contains 20,000 negative and 20,000 positive crack images, enabling image classification tasks. These images have 227 x 227 pixels and are represented in RGB channels.The images exhibit variations in surface finish and illumination conditions, contributing to a diverse and realistic representation of real-world scenarios. To conduct our experiments, we selected a representative part of the dataset. To validate the segmentation output, 200 mask segmentations were created manually. These masks serve as ground truth (GT) references for evaluating the segmentation algorithm.

To evaluate the performance of the proposed approach regarding the classification task, we compute accuracy, precision, F1-score, recall and confusion matrix. To assess the semantic segmentation predicted by the model, we use some metrics such as AUC ROC, AU PRC, DICE and IoU [28].

4.2 Implementation details

We trained all our models using the dataset described during 300 epochs with the Adam optimizer, a learning rate of 0.00001 and batch size of 32. We defined 5 convolutional blocks in the VAE determining 32 as the dimension of the latent coding space. Variable t has a value of 20 in log-barrier loss. The binary cross-entropy loss was computed for segmentation and L2 in classification. The threshold value optimized and used to classify the images was 0.2366. Regarding the hardware, we used NVIDIA RTX 3090 24 GB x 1, 525.60.11 drivers & CUDA 12.0, MSI Z270 Gamming PRO Carbon (MS-7A63); 32 GB and Intel i7-7700K (4.2 GHz), whereas the software used was Pytorch for building and training the models, and Sci-kit learn for evaluation.

5 Results

The results of the experimentation will be discussed below according to the methodology proposed.

5.1 Classification

After the model is trained, we use a not seen 200-image dataset without defects to calculate the error between the images and the reconstructed images ($x - \hat{x}$). We hypothesise that the optimized threshold will have the compromise to assume that those images with reconstruction error above the threshold will be anomalies. To do so, we use the mean error plus the standard deviation error and the result is 0.2366. We plot the histograms (Figure 3, left) to see which is the distribution and where is located the calculated error.

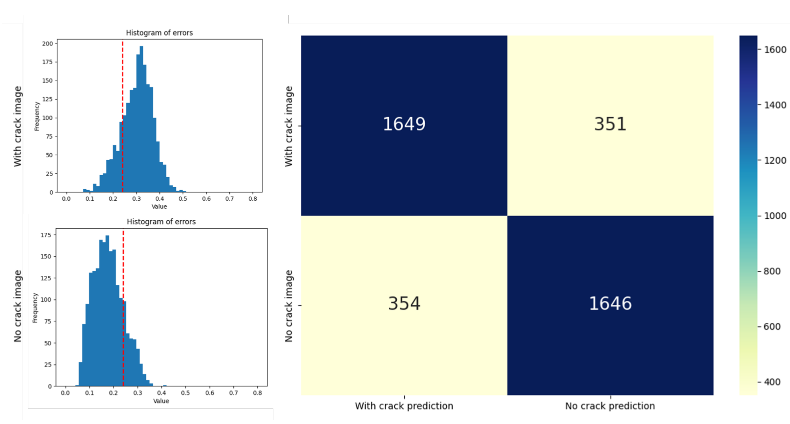

Fig. 3. Histograms of errors in crack images (up left) in images without cracks (down left) and confusion matrix (right).

The confusion matrix is included (Figure 3, right), considering 4,000 test unseen images. Furthermore, metric results to evaluate the classification are considered in Table 1.

Table 1. Metrics for the crack-non crack classification using the optimized threshold.

Accuracy	Precision	Recall	F1-score
0.823	0.823	0.824	0.823

Since there is no previous work using unsupervised methods on the subject, we compare ourselves with supervised studies, bearing in mind that we will not be up to the task. In [29], STCNet I model accuracy over the same dataset is 99.71% and in [18] VGG16 gets an average accuracy of 99.61% using transfer learning.

5.2 Unsupervised Defect Segmentation

To assess the model performance in unsupervised segmentation, we consider an iteration as one full training with a singular seed. The quantitative results by iteration and averaged of AU_ROC, AU_PRC, DICE, IoU are shown in 2.

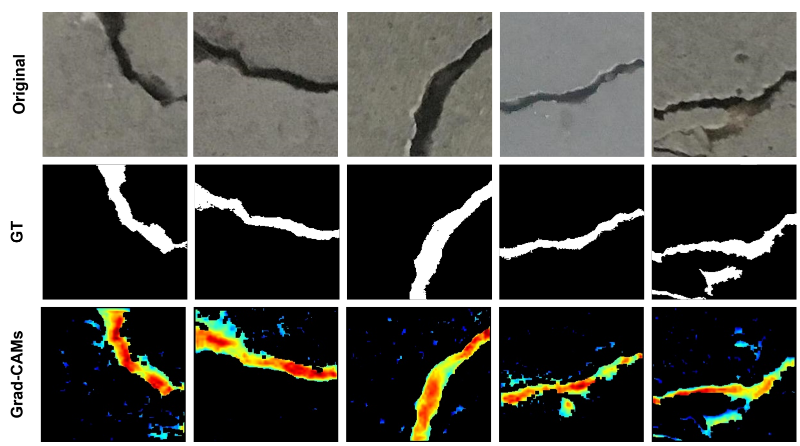

Fig. 4. Qualitative assessments of various examples: original, GT and Grad-CAMs.

Given the absence of prior research utilizing unsupervised methods for this subject, we acknowledge our comparison with supervised studies while acknowledging that we may not achieve the same level of performance. In [30] a CNN combined with ViT shows a IoU score of 0.82 in segmentation using our dataset. They add other supervised models as U-Net and DeepLabv3+ getting a IoU 0.79 and 0.82 respectively.

Table 2. Three Iterations Results over 200 unseen images and Average.

metric	AU_ROC	AU_PRC	DICE	IoU
Iter 0	**0.97**	0.76	0.71	**0.55**
Iter 1	0.95	0.64	0.65	0.48
Iter 2	0.96	**0.78**	**0.72**	0.54
Average	0.96	0.73	0.69	0.53

6 Conclusion

A relevant body of literature on defect detection in the industry requires manual annotation to train deep learning-based models. The annotation process is an expensive task that increases the costs of the industrial process. To overcome this issue, in this work, we propose an unsupervised anomaly detection algorithm able to detect and find the crack location in concrete images. The proposed method based on VAE and Grad-CAMs allows the detection and segmentation of defects showing promising results. Our approach aims to serve as a foundational step towards achieving zero-defect manufacturing, providing a holistic solution to minimize deviations in the operation of buildings and manufacturing processes.

Funding

This work has received funding from Horizon Europe, the European Union's Framework Programme for Research and Innovation, under Grant Agreement No. 101058054 (TURBO) and No. 101057404 (ZDZW). The work of Rocío del Amor has been supported by the Spanish Ministry of Universities (FPU20/05263).

References

[1] A. Gupta, "Current research opportunities of image processing and computer vision," *Computer Science*, 2019, ISSN: 2300-7036, 1508-2806. DOI: 10.7494/csci.2019.20.4.3163.

[2] H. Tian, T. Wang, Y. Liu, X. Qiao, and Y. Li, "Computer vision technology in agricultural automation —A review," *Information Processing in Agriculture*, Mar. 2020. DOI: 10.1016/j.inpa.2019.09.006.

[3] S. Paneru and I. Jeelani, "Computer vision applications in construction: Current state, opportunities & challenges," en, *Automation in Construction*, Dec. 2021. DOI: 10.1016/j.autcon.2021.103940. (visited on 05/12/2023).

[4] H. Taheri, M. Gonzalez Bocanegra, and M. Taheri, "Artificial Intelligence, Machine Learning and Smart Technologies for Nondestructive Evaluation," *Sensors (Basel, Switzerland)*, vol. 22, 2022, ISSN: 1424-8220. DOI: 10.3390/s22114055.

[5] A. Saberironaghi, J. Ren, and M. El-Gindy, "Defect Detection Methods for Industrial Products Using Deep Learning Techniques: A Review," *Algorithms*, 2023, ISSN: 1999-4893. DOI: 10.3390/a16020095.

[6] P. M. Bhatt, R. K. Malhan, P. Rajendran, *et al.*, "Image-Based Surface Defect Detection Using Deep Learning: A Review," *Journal of Computing and Information Science in Engineering*, 2021, ISSN: 1530-9827, 1944-7078. DOI: 10.1115/1.4049535.

[7] L. Bommes, T. Pickel, C. Buerhop-Lutz, J. Hauch, C. Brabec, and I. M. Peters, "Computer vision tool for detection, mapping, and fault classification of photovoltaics modules in aerial IR videos," *Progress in Photovoltaics: Research and Applications*, 2021, ISSN: 1099-159X. DOI: 10.1002/pip.3448.

[8] D. Mery and C. Arteta, "Automatic Defect Recognition in X-Ray Testing Using Computer Vision," 2017. DOI: 10.1109/WACV.2017.119.

[9] G. Fragidis and D. Konstantas, "Customer-Centric Service Design: Featuring Service Use in Life Practices," 2022. DOI: 10.1007/978-3-031-14844-6_15.

[10] D. V. Enrique and A. L. Soares, "Cognitive Digital Twin Enabling Smart Product-Services Systems: A Literature Review," 2022. DOI: 10.1007/978-3-031-14844-6_7.

[11] Y. Tang, Y. Lin, X. Huang, M. Yao, Z. Huang, and X. Zou, "Grand Challenges of Machine-Vision Technology in Civil Structural Health Monitoring," *Artificial Intelligence Evolution*, Mar. 2020, ISSN: 2717-5952. DOI: 10.37256/aie.112020250.

[12] W. Fang, L. Ding, P. E. D. Love, *et al.*, "Computer vision applications in construction safety assurance," *Automation in Construction*, 2020, ISSN: 0926-5805. DOI: 10.1016/j.autcon.2019.103013.

[13] B. F. Spencer, V. Hoskere, and Y. Narazaki, "Advances in Computer Vision-Based Civil Infrastructure Inspection and Monitoring," *Engineering*, vol. 5, 2019, ISSN: 2095-8099. DOI: 10.1016/j.eng.2018.11.030.

[14] W. R. L. d. Silva and D. S. d. Lucena, "Concrete Cracks Detection Based on Deep Learning Image Classification," *Proceedings*, vol. 2, 2018, ISSN: 2504-3900. DOI: 10.3390/ICEM18-05387.

[15] L. Yang, B. Li, W. Li, Z. Liu, G. Yang, and J. Xiao, "Deep Concrete Inspection Using Unmanned Aerial Vehicle Towards CSSC Database," 2017.

[16] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks," *Computer-Aided Civil and Infrastructure Engineering*, 2017. DOI: 10.1111/mice.12263.

[17] D. Ai, G. Jiang, S.-K. Lam, P. He, and C. Li, "Computer vision framework for crack detection of civil infrastructure—A review," *Engineering Applications of Artificial Intelligence*, 2023, ISSN: 0952-1976. DOI: 10.1016/j.engappai.2022.105478.

[18] M. M. Islam, M. B. Hossain, M. N. Akhtar, M. A. Moni, and K. F. Hasan, "Cnn based on transfer learning models using data augmentation and

transformation for detection of concrete crack," *Algorithms*, vol. 15, no. 8, p. 287, 2022.

[19] A. Zhang, K. C. P. Wang, B. Li, *et al.*, "Automated Pixel-Level Pavement Crack Detection on 3D Asphalt Surfaces Using a Deep-Learning Network," *Computer-Aided Civil and Infrastructure Engineering*, 2017. DOI: 10.1111/mice.12297.

[20] L. Pei, Z. Sun, L. Xiao, W. Li, J. Sun, and H. Zhang, "Virtual generation of pavement crack images based on improved deep convolutional generative adversarial network," *Engineering Applications of Artificial Intelligence*, 2021. DOI: 10.1016/j.engappai.2021.104376.

[21] J. K. Chow, Z. Su, J. Wu, P. S. Tan, X. Mao, and Y. H. Wang, "Anomaly detection of defects on concrete structures with the convolutional autoencoder," *Advanced Engineering Informatics*, Aug. 2020. DOI: 10.1016/j.aei.2020.101105.

[22] Z. Rastin, G. Ghodrati Amiri, and E. Darvishan, "Unsupervised Structural Damage Detection Technique Based on a Deep Convolutional Autoencoder," *Shock and Vibration*, Apr. 2021. DOI: 10.1155/2021/6658575.

[23] A. J. Titus, O. M. Wilkins, C. A. Bobak, and B. C. Christensen, "Unsupervised deep learning with variational autoencoders applied to breast tumor genome-wide DNA methylation data with biologic feature extraction," Bioinformatics, Oct. 2, 2018. DOI: 10.1101/433763.

[24] D. Foster, "Generative deep learning: Teaching machines to paint, write, compose, and play," 2019.

[25] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," pp. 618–626, 2017.

[26] J. Silva-Rodríguez, V. Naranjo, and J. Dolz, "Constrained unsupervised anomaly segmentation," *Medical Image Analysis*, vol. 80, p. 102 526, 2022.

[27] Ç. F. Özgenel and A. G. Sorguç, "Performance comparison of pretrained convolutional neural networks on crack detection in buildings," in *Isarc. proceedings of the international symposium on automation and robotics in construction*, IAARC Publications, vol. 35, 2018, pp. 1–8.

[28] R. Mahmoudi, N. Benameur, R. Mabrouk, M. A. Mohammed, B. Garcia-Zapirain, and M. H. Bedoui, "A deep learning-based diagnosis system for COVID-19 detection and pneumonia screening using CT imaging," *Applied Sciences*, Jan. 2022. DOI: 10.3390/app12104825.

[29] W. Ye, S. Deng, J. Ren, X. Xu, K. Zhang, and W. Du, "Deep learning-based fast detection of apparent concrete crack in slab tracks with dilated convolution," *Construction and Building Materials*, vol. 329, p. 127 157, 2022.

[30] E. A. Shamsabadi, C. Xu, A. S. Rao, T. Nguyen, T. Ngo, and D. Dias-da Costa, "Vision transformer-based autonomous crack detection on asphalt and concrete surfaces," *Automation in Construction*, vol. 140, p. 104 316, 2022.