

Automatic detection based on Deep Learning of small cracks in induction thermography using images from FEM Simulation Models

By O. Antero*, A. Muniategui*, E. Gorostegui-Colinas*, A. Gonzalez Conde, B. Oswald-Tranta** and P. Westphal***

* LORTEK, Basque Research and Technology Alliance (BRTA), Arranomendia kalea 4A, 20240, Ordizia, Spain, oaantero@lortek.es

** Chair of Automation, University of Leoben, Peter-Tunnerstr. 27, 8700 Leoben, Austria

*** GKN Aerospace Engine Systems, Trollhättan, Sweden

Abstract

Within industrial applications, the number of available images with cracks is low. This is a limiting factor when developing thermography-based inspection systems that use Deep Learning (DL)-based models for crack detection. Another relevant point is that the annotation of images is time consuming and subjected to variability. Relative to the first limiting factor, generative Artificial Intelligence (AI) can fill this gap. In this paper a method for automatic detection of small cracks in induction thermography that makes use of Generative Adversarial Networks (GANs) for synthetic image generation for model training is introduced. GANs are trained with images of Vastrestraint Test and Finite Elements Method (FEM) simulations. Concerning annotation of images, a fast and robust annotation method is defined. This work shows that the use of synthetic data is suitable for training DL models.

1. Introduction

Induction thermography is an especially useful inspection method for the detection of surface defects comparing to other NDT techniques, such as magnetic particle inspection [1], as it has very low impact on the inspected material, and it is a process that can be automatized [2], [3]. In this kind of inspections, the work samples are heated, and the heating process is captured using an infrared camera. The recorded thermographic measurements undergo various image processing techniques, which can enhance the visibility of defects within the samples. These techniques are selected based on the specific characteristics of the sample and the types of defects that may appear in them. Then, the enhanced thermal images are examined to detect the presence of defects.

In order to automatize an induction thermography-based inspection system one of the primary aspects to automatize is the defect detection task. In this field, some of the previous work shows there are several approaches making use of Convolutional Neural Networks (CNN) [4][5], such as classification, object detection or semantic segmentation [6]. One of the challenges when using this kind of neural networks is usually the need of datasets with numerous samples to train, since normally in industrial settings the number of defective samples is low. Introducing synthetic data to the set of real samples can be an option to overcome this problem [7].

For this paper, different welding samples were selected, which were produced using Vastrestraint Test Machine [8]. This machine applies a certain strain by bending the sample during the welding process, causing small cracks to appear (in the order of the millimetre or smaller). The Vastrestraint samples were measured using induction thermography and then processed by applying Fast Fourier Transform (FFT) with the purpose of obtaining the phase image. In this kind of image, the small cracks present in the samples display a specific pattern, called "butterfly pattern", which is especially helpful for the posterior automatic defect detection.

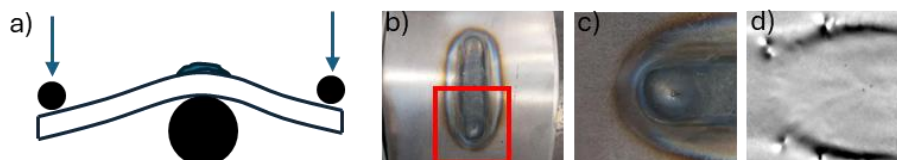


Figure 1. a) Vastrestraint test scheme, b) sample example, c) inspection window, d) phase image.

Apart from this experimental material, for the purpose of generating synthetic data to train automatic detection models, there were also employed some phase images from FEM simulations. The idea of using these images is to prove the possibility of generating synthetic images sufficiently close to the real images so as not to have to rely exclusively on the availability of samples with real cracks to train robust automatic defect detection models.

With all this, in this study it is introduced an approach for the training of Deep Learning based automatic defect detection models for industrial inspection systems based on induction thermography, putting the focus on the detection of the pattern that small cracks define in the phase images. This approach involves image annotation, synthetic data generation making use of Generative Adversarial Networks (GAN), namely pix2pix [9], dataset generation and training of CNN.

This paper is divided as follows. The first section describes the datasets. First, the datasets created from Vastrestraint Test-based images and FEM-based images are described. Then, this section is divided into two sub-sections. In the first subsection, a new annotation method to speed up annotation process and reduce annotation variability is introduced. In the second subsection, synthetic image generation is described. This paper presents a procedure to make easier the training of automatic defect detection models for when the available defective data is reduced, while obtaining acceptable detection results. The second section describes the detection models used. In this section, first the models are described and second, the evaluation method used is introduced. Finally, results obtained are described as discussed.

2. Datasets

In order to analyze if synthetic images are suitable for DL-based model training for butterfly-like pattern detection, different types of datasets were created. Two reference datasets have been used. The first one, a dataset that consists of phase images obtained from FFT transformed thermography recordings of samples obtained from Vastrestraint Test (see **Figure 1.d**). And the second one, a dataset composed of 72 phase images obtained via FEM simulations (see **Figure 2**) [4]. FEM images have been obtained by using the multi-physics simulation package ANSYS15. The model is calculated in a coupled way. First, in the electromagnetic part, the induced eddy currents are computed. Then, based on these induced eddy currents, the Joule heating is determined. In the second step, the thermal process and heat diffusion during and after the heating pulse is simulated. The calculated temporal change of the surface temperature is evaluated to a phase image by Fourier transform. This is done in the same way as for the measured data. Several parameters have been varied in FEM simulations, i.e. defect length, defect depth, orientation, inclination angle and shape of the cracks.

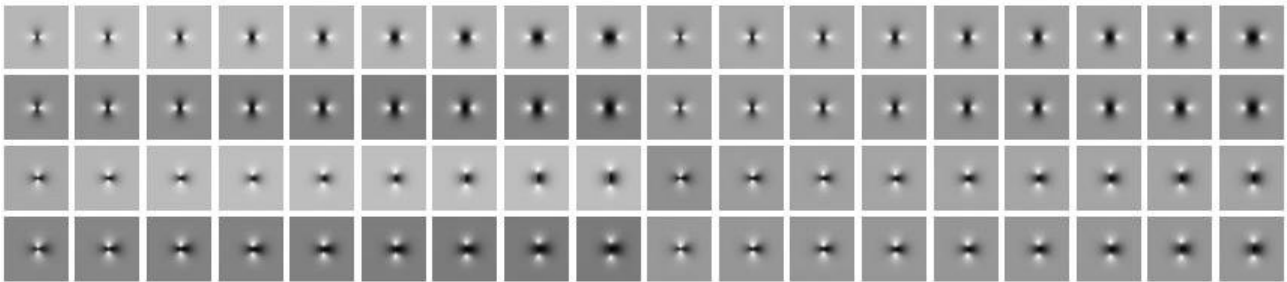


Figure 2. Input 72 simulated cracks.

Vastrestraint Test-based dataset consists of approximately 200 images. This dataset has been divided into two sub-datasets. One of these sub-datasets is used as reference dataset, with real phase images, for benchmarking of trained models, as well as for DL-based model training. In this respect, this dataset is divided into train, validation and test. First two are used to train DL-based models for crack detection based only on real phase images. Results of models trained with this dataset will represent the reference to which other models trained with alternative dataset will be compared to. The remaining sub-dataset is used to train pix2pix model. And hence, to create synthetic realistic phase images.

FEM-based images that consisted of 72 images are augmented to 1000 using data augmentation, i.e. rotation, shear, translation etc. Obtained images are then combined at random to obtain images with more than one crack (see **Figure 3**). These images are used to create a dataset for pix2pix model training to generate synthetic images. Synthetic image creation for Vastrestraint Test-based images and FEM-based images is described in a following section.

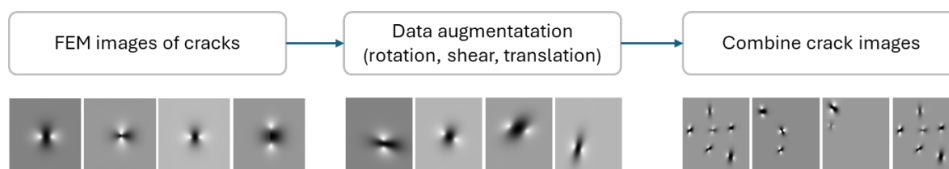


Figure 3. Process to create combined images from FEM simulation images.

Vastrestraint Test-based and synthetic images obtained from Vastrestraint Test and FEM-based images has been combined as indicated in **Table 1**. In total 10 datasets were created. In this table, V is used to denote Vastrestraint Test and F to denote FEM. Additionally, s is used to refer synthetic images. This way, V100 dataset is composed of 100 samples from Vastrestraint Test, sF100 dataset consists of 100 images obtained via pix2pix trained with combined FEM images, and

V200-sF100 contains images from dataset V200 and sF100. All models trained using these datasets were tested in Vareststraint Test-based images, images not included neither in V100 nor V200.

DATASET NAME	# IMAGES		
	VARESTRAINT	SYNTHETIC FROM VARESTRAINT	SYNTHETIC FROM FEM
V100	100	-	-
V200	200	-	-
sF100	-	100	-
sF200	-	200	-
sV100	-	-	100
sV200	-	-	200
V200-sF100	200	100	-
V200-sF200	200	200	-
V200-sV100	200	-	100
V200-sV200	200	-	200

Table 1. Types of datasets defined.

2.1. Annotation

Annotation of the images shown in **Figure 1** and **Figure 2** could be carried out by determining the whole perimeter of each of the blobs – two black and two white blobs per butterfly pattern – or the whole perimeter of each crack. This annotation method is useful for segmentation model training. One of its drawbacks is that it is time consuming and that it has an inherent variability due to the difficulty to differentiate the boundary between black regions and background or white regions and background. Although this annotation could be automated for FEM images, it could continue to be complex for real phase images.

It has to be considered that for many industrial applications, the detection of crack is itself sufficient and hence, the detection of the whole perimeter of the butterfly pattern is not required. Furthermore, in cases where the size of the crack needs to be determined, an approximate value of its size can be enough. Thus, simpler annotation methods could be better. In this work, butterfly patterns have been annotated as follows (see **Figure 4**): one point for the center of the crack and one point for each of the centers of the blobs of the crack (two black and two white). The center of the crack can be used to train detection models for Hit/Miss type of detection. While the centers of blobs can be used to train models directed to identify approximately the size of the detected crack. This annotation method is faster, and it has more reduced variability than segmentation-based ones. This comparison it is not included in this work.

Vareststraint Test-based images and FEM-based images were annotated using this method in COCO Annotator tool [10]. Annotated data was then used to create synthetic images with their associated Ground Truth (GT).

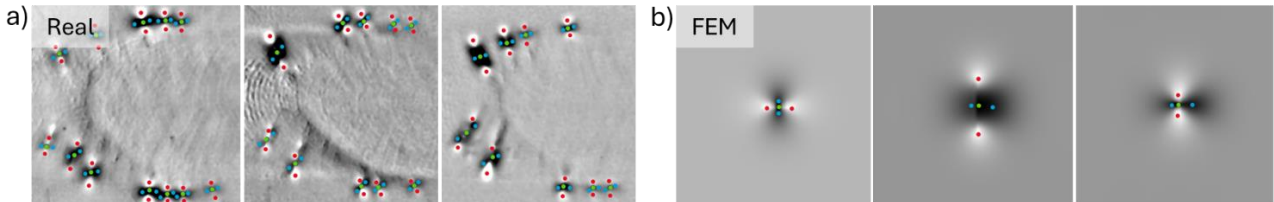


Figure 4. Examples of the alternative annotation method used.

2.2. Generation of synthetic images

The generative model chosen to create synthetic data was pix2pix. This model is a conditional GAN (cGAN) [11] as its basis. This means that the generation of data is conditioned by some additional data. In this case the data provided to train pix2pix, apart from the images created in the previous step, are the labels of those images in the form of a binary mask. Thus, the model learns to recreate more precisely the relevant information of the input image. After training, the model can generate the synthetic data from a label. Hence, each generated synthetic image has its corresponding GT or annotation. This reduces the time required for synthetic image annotation, i.e. a visual inspection for GT correction is generally enough.

To train the GAN there were 2 starting points: the images from FEM simulation models and the images from Vareststraint test samples. In the case of the first ones, images obtained by combining FEM images were used to ensure a higher variability in the set of images to feed the pix2pix (see previous section). In the case of the second ones, no additional pre-processing was carried out.

After pre-processing of images, both types of images (combined images of simulated FEM cracks and real crack images) were used following the same procedure (see **Figure 5**). First, due to the need of annotated masks within the input of pix2pix, binary masks were created from annotated points. Note that since the center of each crack, as well as the centers of their blobs were annotated, different types of masks could be created from annotations. For instance, circles with predefined radius in pixels or ellipses with the sizes of both edges proportional to the distance between the centers of black and white blobs. In this second case, the ellipses define not only the position of the crack, but also the size and orientation. Once GT masks were obtained, pix2pix was trained for each kind of input images (Varestraint Test and FEM images), resulting in 2 generative models capable of generating synthetic data from labels (see **Figure 5.b**). Synthetic images and their associated GT masks for FEM and Varestraint Test images are shown in see **Figure 5.c** and **d** respectively.

As a final step, generated synthetic images are visually inspected to check that the GT mask of each synthetic image represents its annotation properly.

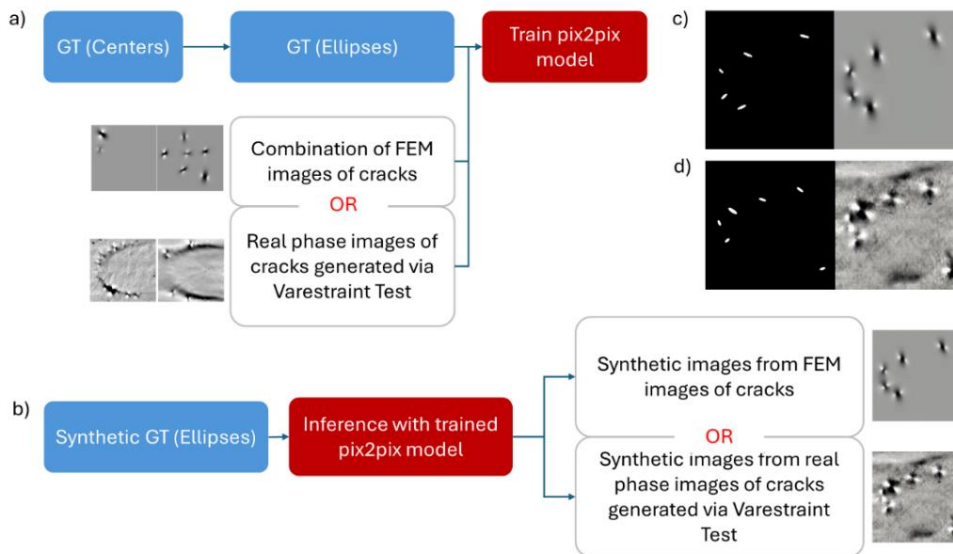


Figure 5. Steps of *pix2pix* model training (a) and generation of synthetic images from trained *pix2pix* models (b).

3. Detection models

As indicated before, from annotated images, cracks can be represented in different ways, for example, as points, as circles with fixed size, and as ellipsis with edges proportional to the distance between the centers of black and white blobs. As a first approximation points and circles were considered for crack representation and DL-based object detection and segmentation models were trained. The size of input images used was of 256 x 256.

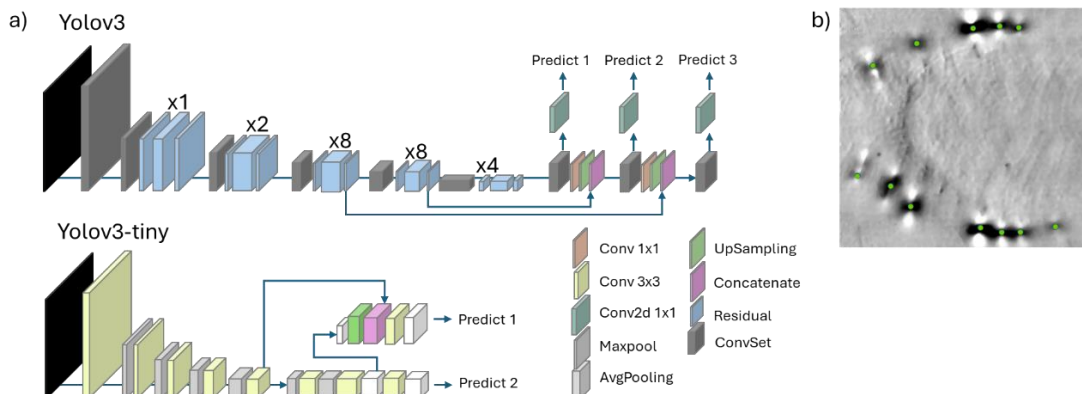


Figure 6. a) architecture of YOLOv3 and YOLOv3-tiny networks and b) type of input, cracks are considered as points.

For object detection YOLO [12] was used, as its architecture is designed to perform real-time object detection, making it suitable for quick detection. In this case the used versions were YOLOv3 [13] and YOLOv3-tiny [14]. This object detection models aims at detecting the object as a box. In this case, annotations were considered as points. In order to make YOLO models suitable to detect objects as points, few modifications were made on the last layers and the loss of

these models. For instance, the number of values to predict was reduced from 5 ($x, y, w, h, score$) to 3 ($x, y, score$) and the loss was modified accordingly. These networks have Darknet-53 and Darknet-tiny as the backbone network. Due to the low number of images in each dataset, the training was carried out by using pretrained models, in where the weights of Darknet were fixed as not trainable and only the weights of the layers of the head of the net were trained. The architecture of YOLOv3 and YOLOv3-tiny is shown in **Figure 6** and the number of trainable, non-trainable and total number of parameters are shown in **Table 2**.

	Number of parameters		
	Trainable	Non-trainable	Total
yolov3-tiny	5,104,664	6,312,816	11,417,480
yolov3	5,405,220	40,637,536	46,042,756
VGG19 _{dec} - UNet[n=8] _{enc}	1,481,889	20,025,344	21,507,233
VGG19 _{dec} - UNet[n=32] _{enc}	11,143,809	20,028,224	31,172,033

Table 2. Number of parameters for each of the trained models.

For the segmentation task U-Net [15] was considered, as it is a neural network designed specifically for that. It has an encoder-decoder structure with skip connections, meaning that in the decoder part it combines upsampled features with corresponding high-resolution features from the encoder. In this case, annotations were transformed to binary masks by replace its annotated point by a circle of fixed size (radius of 5 pixels in size) centered on each point. Comparing to object detection models, segmentation models need less input data to train robust models, while giving a more precise detection. For this study, U-Net was trained with a VGG19 network in the encoder part [16]. In the following, this model will be referred as VGG19_{enc}-UNet_{dec}. The architecture of the model is shown in **Figure 7**. Similarly to YOLO models, this model is trained by using a VGG19 model pretrained in Imagenet, fixing the encoder weights and set as trainable only the decoder part of VGG19-UNet_{dec}. Additionally, two encoder sizes were considered being the number of filters of its layers the following ones: [128, 64, 32, 16, 8] for an small architecture and [512, 256, 128, 64, 32] for a large architecture. The small architecture will be referred in the following as VGG19_{enc}-UNet[n=8]_{dec} and the large architecture ad VGG19_{enc}-UNet[n=32]_{dec}. The number of trainable, non-trainable and the total number of parameters is indicated in **Table 2**.

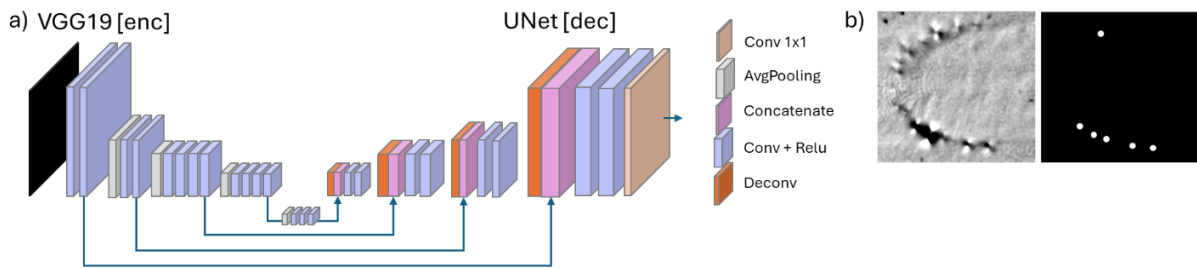


Figure 7. a) architecture of U-Net network and b) type of mask used to represent annotations.

3.1. Data-augmentation: noise

As mentioned before, some of the detection models are trained using uniquely synthetic images, and these images do not have the same background as the images of real samples (see **Figure 8.a**). This can result in a model that is not capable to distinguish the cracks from the background noise. Therefore, in addition to usual data-augmentation options (shear, crop, saturation-contrast-hue changes, flips, rotations etc), the option of adding Perlin, fractal (which is a variation of Perlin) [17] and Gabor noise [18] as were added to data-augmentation during model training. The type of added noise for different degrees are shown in **Figure 8.b**. Some patches of the background noise from real images are shown in this image to compare them with added noises. Perlin noise is a type of gradient noise and one of its uses is to create image texture. In this case it was used to add random noise to the background of the synthetic images during training. Fractal noise is similar to Perline noise, and it is also denoted as type of Brownian noise. Finally, Gabor noise is defined as the convolution of a Poisson point distribution with the real part of a Gabor kernel.

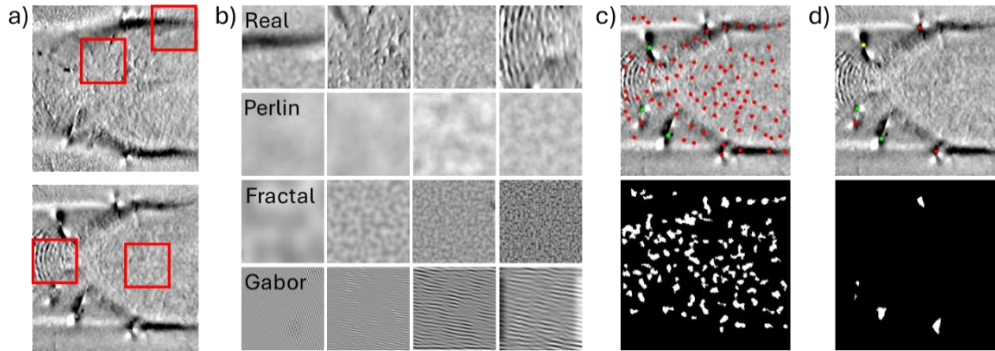


Figure 8. a) real sample images, b) noises applied as data augmentation to recreate the real background, c) U-Net detection without applying noise data augmentation, d) U-Net detection after applying noise data augmentation

As an example of the effect of adding noise within data augmentation, results for a VGG19_{enc}-UNet_{dec} model are shown in **Figure 8**. When models were trained without adding noise within data augmentation, many False Positives (FP) were detected (see **Figure 8.c**). In contrary, these FP were considerable reduced in the predictions of models trained adding indicated noise within data augmentation (see **Figure 8.d**).

3.2. Model evaluation

In order to make the results of both models comparable, the same evaluation method was used. For instance, a detection is considered to be a True Positive (TP) if the predicted point is at a distance lower than a predefined threshold from the annotated point. The threshold was set to 20 pixels. In the case of YOLO models, the evaluation is straightforward since its output correspond to the coordinates of the predicted points. In the case of VGG19_{enc}-UNet_{dec}, the output was post-processed to transform the binary mask to a list of points. This is carried out by applying connected components to the output mask, keeping those predictions with more than 10 pixels and by determining their center of mass that will be the predicted points.

Both YOLO and U-Net were trained with the datasets described in Table x, and even if some of the training datasets only contain synthetic images, all the models were evaluated using the same images from real samples. In the case of U-Net the evaluation method was changed to do it the same way as YOLO, so this way the obtained results are easily comparable.

4. Probability of detection

The probability of detection (POD) is a statistical method to study the detectability and reliability of an inspection system, usually applied to NDT techniques [19]. There exist two kinds of POD analysis: hit/miss and \hat{a} vs. a . The first one evaluates detection performance analyzing whether a defect has been detected or not, and the second one, compares the POD as function of a signal (\hat{a}) versus a threshold (a) to assess the effectiveness of the detection system. In this case a hit/miss analysis has been carried out, with which it has been measured the reliability and accuracy of the detection with each trained model, obtaining the minimum length that can be detected confidently. The detectability parameters usually taken are 90% POD and 95% confidence interval, being a_{90} and $a_{90/95}$ the cracks size related to each one. The latter refers to a statistical tool about the data distribution. There are 139 cracks present in the set used to perform POD analysis, in a size range between 0.17mm and 3.15mm with the distribution shown in **Figure 1Figure 9**.

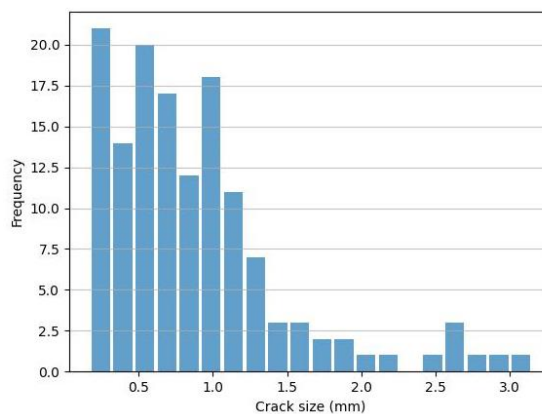


Figure 9. Distribution of the cracks by length

Very briefly, the steps followed for the POD analysis have been:

1. Detection of all cracks using liquid penetrant testing (PT) as NDT of reference
2. Inspector A marks all visible cracks in the phase images
3. Training of the algorithms using real and synthetic images
4. Manual marking of the hit/miss for the automatic detection of each model with respect to the PT reference
5. Calculation of POD curves and parameters [20], [21]

This way, it is possible to evaluate not only the detection accuracy obtained after training the automatic detection models, but also the reliability of thermography as non-destructive inspection system.

5. Results

In order to evaluate the feasibility of the models trained using synthetic data, in the following lines the results obtained from each trained model are shown. A comparative analysis has been carried out based on the Average Precision (AP)[22] of the performance of the trained models. AP is a metric that evaluates the accuracy of a model by combining precision and recall into a single metric. The accuracy results obtained for each kind of dataset described in the previous section are summarized in **Figure 10**, specifying the result each kind of model has given.

Observe that in almost every dataset the model with better performance is YOLOv3. Furthermore, even though when training using only synthetic datasets the results are lower than when training with datasets with real cracks information, it is possible to manage an accuracy of 75%, and when adding these images to real images the performance of the models increases around 3-4% comparing to the result of the models only trained with real data. Observe that the results improve when adding more trainable parameters: results for YOLOv3 are a bit better than those of YOLOv3-tiny, and, in general, results improve for of VGG19_{enc}-UNet[n=32]_{dec} compared to of VGG19_{enc}-UNet[n=8]_{dec}.

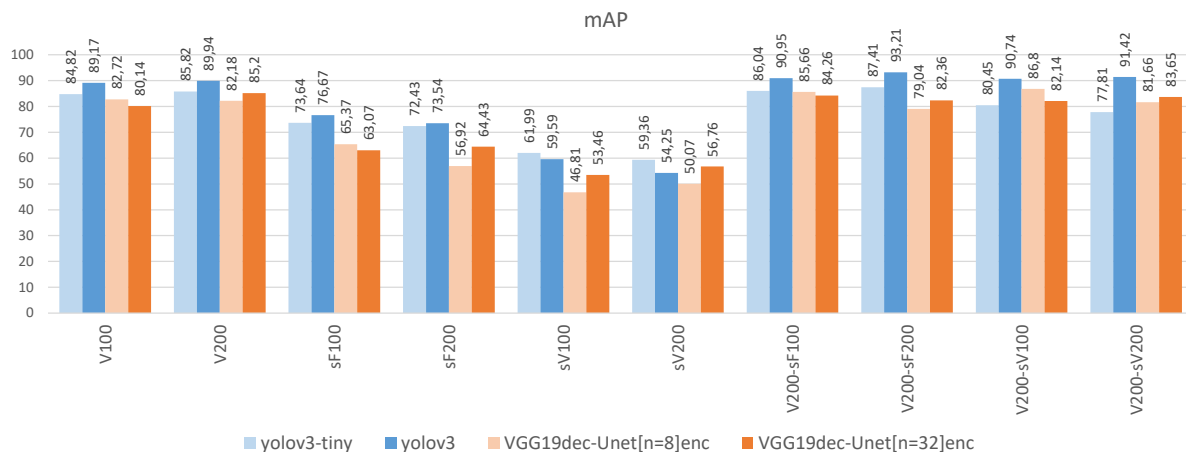


Figure 10. AP results for each trained dataset, separated by type of detection model.

Besides these results, it has been evaluated the effect of adding noise as data augmentation in the training of the datasets that only contain synthetic images generated from FEM simulation images. As it can be seen in **Figure 11**, in almost all the studied cases the accuracy of the models increased by 15% or more when trained using the noises detailed in section 3, irrespective of whether the network used is YOLO or U-Net. This difference is higher for of VGG19_{enc}-UNet_{dec} models.

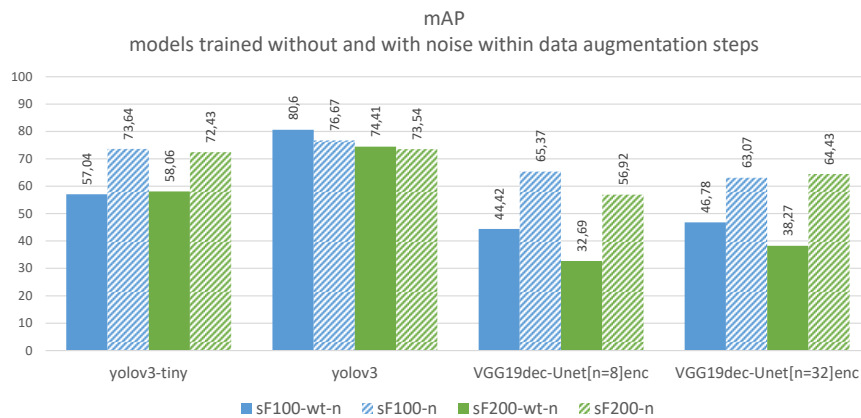


Figure 11. AP results for each type of detection model trained only with synthetic images, with and without adding noise as data augmentation.

After having obtained all the results from the models, a Hit/Miss analysis has been carried out by plotting the POD curves, as a way of comparing the results obtained with YOLO and U-Net models for each trained dataset. The obtained POD curves are shown in **Figure 12**. In each graph the POD curves with a close-up box showing the values of a_{90} and $a_{90/95}$ is shown. These values are also represented in the legend on the graphs to easy visualization of POD results.

The aim of carrying out POD analysis has been two-fold. First, to determine whether the use of synthetic images for model training is feasible, i.e. obtained results are similar to those obtained with real phase images. Second, to analyze the effect of the number of images used for model training in the reliability of the detected cracks. The results of the first analysis are shown for U-Net in **Figure 12.a**, and for YOLO in **Figure 12.b**. Both are compared using the datasets containing 200 images. The results of the second analysis, are shown for U-Net in **Figure 12.c**, and YOLO in **Figure 12.d** models trained with datasets with 100 images and 200 images. In all of these cases the reference of the detection made by the inspector (GT) is added since it is the reference to evaluate how good the automatic detection is.

The results in the graphs show that the minimum crack length detected reliably is around 0.94mm and 1.14mm in the majority of the cases. The values are higher with YOLO models than with U-Net, except when training with the dataset of 200 synthetic images generated from Vareststraint test samples. Also, it is important to mention that in **Figure 12.c**, the U-Net sV100 model does not meet the requirements to consider a correct POD analysis. This is because the number of Misses we have is too high when we look for a $POD > 0.9$. Results show that the use of synthetic images generated from FEM simulations are reliable for crack detection on phase images of inductive thermography.

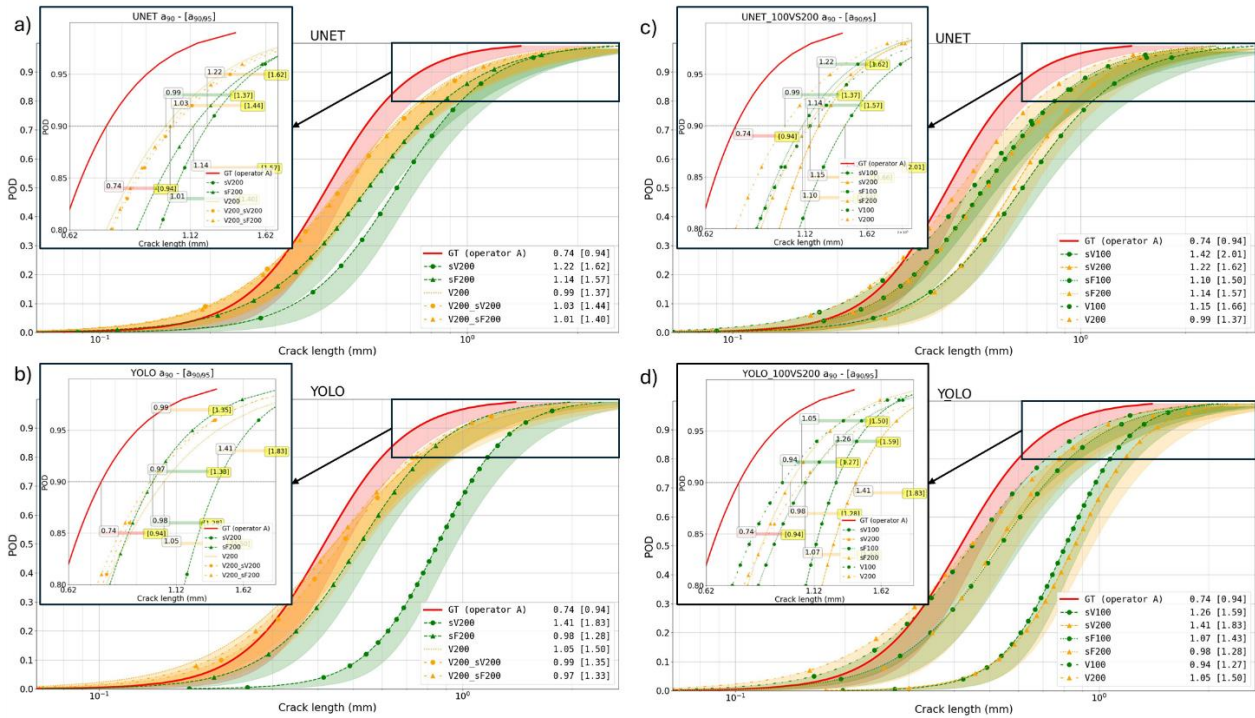


Figure 12. Comparison of POD curves of hit/miss analysis a) U-Net with all the datasets with 200 images, b) YOLO with all the datasets with 200 images, c) U-Net with 100 and 200 training images, d) YOLO with 100 and 200 training images. The graph added in black box in each graph is a zoomed region of the POD graph and it is used to indicate the a_{90} and $a_{90/95}$ values.

6. Conclusions

In this paper a method for defect detection based on DL algorithms for small cracks in induction thermography inspection based on FEM images is shown. The objective of this study was twofold. First, simplify the annotation of the images, reducing required time for annotation and reducing variability. Second, it is shown that the use of synthetic images it is a good alternative for robust model training. Additionally, the addition of different types of noise in data augmentation steps, when training models with this data, has showed to improve the results considerable.

So as for the defect detection models, the results of this study have proved that the usage of synthetic images can be beneficial for this task, as it improves the accuracy in the majority of the tested cases. What is more, when training DL-based detection models uniquely with synthetic data generated from FEM simulations it has showed that the detection accuracy can reach up to 75%, making it promising for many applications. For example, in the early stages of the automatization of an induction thermography-based inspection system, when there are hardly any defective samples.

When it refers to the hit/miss analysis, the detection models were evaluated not only by means of the accuracy obtained, but also by analyzing the probability of detecting a crack according to its size, obtaining a minimum size of crack that it is possible to detect reliably when using induction thermography and automatic defect detection models. In any case, it is important to highlight that the value of the algorithm lies in the Hit/Miss ratio and the $a_{90/95}$ value with respect to the technology with which it has been trained, as this is the one the algorithm aims to replace. In this case, Inspector A, as it is the reference used to train. That is, if 100 cracks have been detected using liquid penetrants and the algorithm detects 58, the algorithm's capability is unclear until the accuracy obtained with Inspector A is known.

REFERENCES

- [1] E. Motukisi, "Magnetic Particle Inspection: Characterisation of the magnetic field for various magnetization techniques," 2012, Accessed: May 30, 2024. [Online]. Available: www.saiw.co.za
- [2] E. Gorostegui-Colinas, R. Hidalgo-Gato, P. L. de Uralde, B. U. Marco, and A. M. Merino, "Induction thermography-based inspection of EBW and TIG welded Inconel 718 components: steps towards industrialization," <https://doi.org/10.1117/12.2558265>, vol. 11409, pp. 98–117, May 2020, doi: 10.1117/12.2558265.
- [3] P. Venegas, E. Ivorra, M. Ortega, and I. S. de Ocariz, "Towards the Automation of Infrared Thermography Inspections for Industrial Maintenance Applications," *Sensors 2022, Vol. 22, Page 613*, vol. 22, no. 2, p. 613, Jan. 2022, doi: 10.3390/S22020613.

- [4] B. Oswald-Tranta, P. Lopez de Uralde Olavera, E. Gorostegui-Colinas, and P. Westphal, "Convolutional neural network for automated surface crack detection in inductive thermography," *Proc. SPIE 12536, Thermosense: Thermal Infrared Applications XLV*, 125360L (12 June 2023), <https://doi.org/10.1117/12.2663485>
- [5] K. Tout, N. Samet, and P. Bouteille, "Automated defect detection on inductive thermography images using supervised and semi-supervised Deep Learning methods," *e-journal of nondestructive testing*, vol. 28, no. 9, Sep. 2023, doi: 10.58286/28510.
- [6] S. Lugin, D. Müller, M. Finckbohner, and U. Netzelmann, "Automated surface defect detection in forged parts by inductively excited thermography and magnetic particle inspection," *Quant Infrared Thermogr J*, Oct. 2023, doi: 10.1080/17686733.2023.2266901.
- [7] Q. Fang, C. Ibarra-castanedo, and X. Maldague, "Automatic Defects Segmentation and Identification by Deep Learning Algorithm with Pulsed Thermography: Synthetic and Experimental Data," *Big Data and Cognitive Computing 2021, Vol. 5, Page 9*, vol. 5, no. 1, p. 9, Feb. 2021, doi: 10.3390/BDCC5010009.
- [8] J. Andersson, J. Jacobsson, and C. Lundin, "A Historical Perspective on V restraint Testing and the Importance of Testing Parameters," *Cracking Phenomena in Welds IV*, pp. 3–23, 2016, doi: 10.1007/978-3-319-28434-7_1.
- [9] M. Tahmid, M. S. Alam, N. Rao, and K. M. A. Ashrafi, "Image-to-Image Translation with Conditional Adversarial Networks," *Proceedings of 2023 IEEE 9th International Women in Engineering (WIE) Conference on Electrical and Computer Engineering, WIECON-ECE 2023*, pp. 468–472, Nov. 2016, doi: 10.1109/WIECON-ECE60392.2023.10456447.
- [10] J. Brooks, "COCO Annotator." 2019. [Online]. Available: <https://github.com/jsbroks/coco-annotator/>
- [11] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," Nov. 2014, Accessed: May 28, 2024. [Online]. Available: <https://arxiv.org/abs/1411.1784v1>
- [12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, pp. 779–788, Jun. 2015, doi: 10.1109/CVPR.2016.91.
- [13] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," Apr. 2018, Accessed: May 24, 2024. [Online]. Available: <https://arxiv.org/abs/1804.02767v1>
- [14] P. Adarsh, P. Rathi, and M. Kumar, "YOLO v3-Tiny: Object Detection and Recognition using one stage improved model," *2020 6th International Conference on Advanced Computing and Communication Systems, ICACCS 2020*, pp. 687–694, Mar. 2020, doi: 10.1109/ICACCS48705.2020.9074315.
- [15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", Accessed: May 09, 2024. [Online]. Available: <http://mb.informatik.uni-freiburg.de/>
- [16] M. Alfarhan, M. Deriche, and A. Maalej, "Robust Concurrent Detection of Salt Domes and Faults in Seismic Surveys Using an Improved UNet Architecture," *IEEE Access*, vol. 10, pp. 39424–39435, 2022, doi: 10.1109/ACCESS.2020.3043973.
- [17] K. Perlin, "Improving Noise".
- [18] B. Galerne, A. Lagae, S. Lefebvre, and G. Drettakis, "Gabor noise by example," *ACM Trans Graph*, vol. 31, no. 4, Jul. 2012, doi: 10.1145/2185520.2185569.
- [19] B. Oswald-Tranta, A. Hackl, P. Lopez de Uralde Olavera, E. Gorostegui-Colinas, and A. Rosell, "Calculating probability of detection of short surface cracks using inductive thermography," *QIRT Journal*, 2022, <https://doi.org/10.1080/17686733.2022.2152259>
- [20] C. Annis, "Statistical best-practices for building Probability of Detection(POD) models, R package mh1823, version 5.4.5." 2018. [Online]. Available: <https://statistical-engineering.com/mh1823/>
- [21] C. Annis, "MIL-HDBK-1823A, Department of Defense Handbook: Non Destructive Evaluation System Reliability Assesment," 2009, Accessed: Jun. 14, 2024. [Online]. Available: <http://assist.daps.dla.mil>.
- [22] A. Anwar, "What is Average Precision in Object Detection & Localization Algorithms and how to calculate it?" [Online]. Available: <https://towardsdatascience.com/what-is-average-precision-in-object-detection-localization-algorithms-and-how-to-calculate-it-3f330efe697b>