






Automated detection of subsurface defects using active thermography and deep learning object detectors

Darío G. Lema , Oscar D. Pedrayes , Rubén Usamentiaga , Pablo Venegas  and Daniel F. García 

Abstract—The need for improved quality control in industry makes object detection crucial. This work addresses the challenging problem of subsurface defect detections using a combination of active thermography and deep learning. The novel contribution of this work is to pose the problem as one of object detection rather than semantic segmentation or classification. The images used as input for the deep learning algorithms are three-channel color images obtained using Principal Component Thermography (PCT). The use of these images improves the signal-to-noise ratio (SNR). A framework to label ground truths automatically is also created. The most widely used deep learning detector algorithms were evaluated and YOLOv5 was selected because of its excellent average precision (AP) and its low inference time. The resulting combination of this algorithm and active thermography is effective and accurate in detecting subsurface defects.

Index Terms—Active infrared thermography, Non-destructing testing, Subsurface defects, Carbon fiber sheet, Principal Component Thermography, Deep learning.

I. INTRODUCTION

IMPROVED quality control is a growing demand in the industry, and is therefore the object of many works. Many of these works are focused on the identification of surface defects [1]–[4]. Many subsurface defects cannot be detected by the human eye, therefore their correct detection represents a challenging problem, vital for the quality control of many different industrial parts.

Several strategies can be used to detect subsurface defects, two of the most common being X-ray and infrared thermography (IRT). X-rays are more accurate than IRT, but less safe [5]. The advantages of infrared thermography are that it is non-invasive, does not need to be in direct contact with the specimen and does not have the harmful radiation effects present in X-rays [6]. Many works have been carried out using IRT in order to detect subsurface defects in non-destructive testing [7]–[9].

There are two types of IRT: passive and active. Passive thermography consists of measuring the temperature contrasts of an object which is involved in a spontaneous heat flow process [10], without adding an external heat source. In active thermography, the specimen is subjected to an external heat source to create a temperature contrast. Inert objects do not exhibit temperature variations, therefore it is necessary to use active thermography in order to detect their subsurface defects. The heat flow within the specimen depends on the properties of the material [11]: the induced flow will be affected in areas with subsurface defects and the areas around them. Anomalies

can be detected by measuring the infrared radiation from the external surface with a thermographic camera.

Usually, the inspection is performed by a skilled operator [12], which makes the detection process subjective and time consuming. For example, in [13] subsurface defects were detected in a bicycle frame. After applying active thermography it was necessary for an operator to review the captured images manually. Since some of the subsurface defects were very subtle, after obtaining the infrared images and applying a post-processing technique called Principal Components Analysis (PCA), it was necessary to improve the signal-to-noise ratio (SNR) content of thermographic data.

Traditionally, detecting subsurface defects was done manually, which made the quality process dependent on the judgment of the operator. To solve this problem, IRT and deep learning can be combined. Deep learning can automate the inspection of different industrial parts quickly and objectively, based on data obtained by active IRT.

The combination of deep learning and IRT has already been used in several fields. One of these fields is the medical, more specifically breast cancer screening. In [14], a two-part model composed of a pre-trained Inception V3 model and a SVM classifier is used in conjunction with IRT to detect breast cancer. In [15], a LeNet-based CNN is used to classify breast cancer images automatically. And in [16], a segmentation network with an autoencoder architecture is used to automatically segment breast cancer areas in IRT images. Another area where IRT and deep learning have been combined is in behavioral monitoring. In [17], an infrared image-based classifier is designed for an Ambient Assisted Living (AAL) system to assist people with disabilities. A dataset of six classes composed of everyday actions, such as walking or sitting on a chair, was also created. The proposed model achieves an accuracy of 87.44%. In [18], extremely low-resolution thermal images are used in combination with a LSTM-based network to classify different human actions. Quality control, is another field where the combination of deep learning and IRT has achieved good results. In [19], deep learning techniques are applied to detect subsurface defects using active IRT, however, only a single image classification technique is used. In [20], a classification model is created to classify subsurface defects in a laminate of glass-fiber-reinforced polymer, achieving an accuracy of 84.03%, precision of 87.62% and recall of 82.43%. In [21], semantic segmentation is combined with IRT to generate a mask of possible internal defects in a carbon fiber part. After testing

several strategies to obtain the semantic segmentation mask, it is concluded that the best performing models are a 3-layer LSTM network and U-Net pre-trained with VGG-16. In [22], a spatio-temporal 3D network is created to detect the presence and depth of subsurface defects in carbon fiber reinforced polymer (CFRP) materials. Its main drawback is the inference time: 4.4 seconds, much higher than the time achieved in this work. In [23], a custom deep learning network is created to detect subsurface defects of steel members in a steel truss bridge. Its conclusion is that the position of the sun can affect the inspection task, since the side of the bridge that is exposed to the direct radiation of the sun can be altered. In [24], active thermography is used together with a VGG-based network to detect cracks in Electron-Beam Welding (EBW) and Tungsten Inert Gas (TIG) weldings. The conclusion is that its CNN can be trained with around 1000 samples, getting good accuracy.

All these cited works have made important contributions, however, there is still room for improvement. The problem of detecting subsurface defects can be posed as an object detection problem, unlike in previous works, where it is posed as a classification and segmentation problem. In image classification the objective is to predict the class of a given image. In the context of this work, the classifier should determine whether it contains defects or not. The aim of object detection is to locate objects of interest using bounding boxes, in this case defects. Semantic segmentation, like object detection, locates the objects of interest. The main difference with object detection is that in semantic segmentation, each pixel of the image is classified. This implies two disadvantages: 1) to annotate a dataset it is necessary to classify all the pixels of the images, while for object detection it is simply necessary to label the objects of interest, and 2) it is not possible to count the detected objects, since it works at pixel level.

The object detection approach, combined with active IRT has not yet been widely explored. Furthermore, the labeling process of the ground truth of a detection problem is less time-consuming than in semantic segmentation [25]. In semantic segmentation a mask containing the defects detected is obtained, while in object detection, the bounding boxes of the defects are obtained [26]. The mask is not needed since the most relevant information is the localization of the subsurface defects.

To improve the quality control of industrial parts, four of the most widely used non-proprietary object detectors are evaluated: SSD [27], YOLOv3 [28], YOLOv4 [29] and YOLOv5 [30]. In order to reduce the time needed, the entire process from data collection to the detection of the subsurface defects, is automated. Usually, the most time-consuming task is data labeling, for which a new framework has been developed. Using this framework, the network used can be quickly trained with the data needed to carry out the inspection of a certain industrial part.

In this work, a carbon fiber sheet with a series of known subsurface defects is used. To deal with the limited amount of data, the part is rotated, in such a way that several images are taken. To get more information, data augmentation is applied to the training data. Thanks to these contributions, the quality control process is significantly improved, and manufacturing

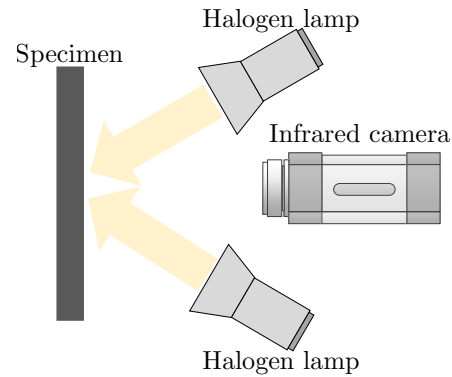


Fig. 1. Infrared inspection using optical step heating (OSHT).

defects can be prevented, lowering costs and increasing safety.

Infrared radiation provides useful information about the temperature of an object. By using infrared measuring devices, it is possible to transform the infrared radiation emitted by an object to an electronic signal. Then, the infrared images can be converted to color images by assigning a color to each level of energy. This false color image is called a thermogram [11]. Thermograms are used in this work to detect subsurface defects.

To apply IRT as a Non-Destructive Testing (NDT) technique it is necessary to supply extra heat to the specimen, typically by means of a heat gun, flash lamp or halogen lamp. The heat gun is the simplest since no setup is needed. Flash lamps and halogen lamps can also be used to heat the specimen accurately. In this work, optical step heating (OSHT) is used with two 1000 W halogen lamps (Eurolite PAR-64 Profi floorspot model), shown in Fig. 1. The lamps are turned on for 10 seconds and then turned off.

Usually, the infrared images captured present noise that hinders the task of defect detection. To deal with this, several post-processing techniques can be applied such as statistical moments, principal component analysis, dynamic thermal tomography, and polynomial fit and derivatives. Principal Component Analysis (PCA) is a static technique to reduce the number of variables of the data and highlight its differences and similarities. As concluded in [11], PCA is the most appropriate method, although it depends on the properties of the defects.

II. MATERIALS AND METHODS

A. Dataset

The dataset is obtained by applying active thermography to a single carbon fiber solid laminate sheet, shown in Fig. 2a. This type of material consists of several carbon fiber layers, each layer with fibers oriented in a specific direction, stacked and bonded together with a resin. Internal defects were induced by means of small sheets of polytetrafluoroethylene (PTFE) and metal shavings. The PTFE simulates delaminations, which are defects produced by the separation of adjacent plies, while the metal shavings simulate accidental inclusion of small pieces of the cutting tools used in the manufacturing process. The twelve subsurface defects are distributed as shown in Fig. 2b.

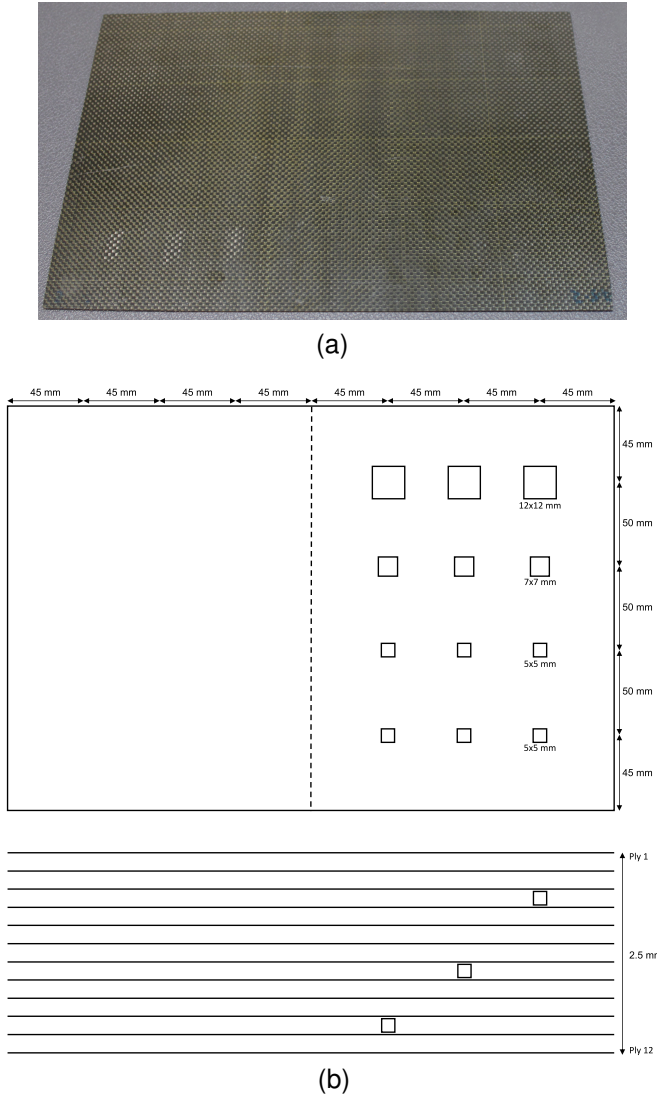


Fig. 2. (a) Carbon fiber part used in this work. (b) Defects present in the carbon fiber part used in this work. The first nine defects are PTFE defects, while the last three are metal shavings. The inspection side is the upper side.

The defects have different sizes: 12×12 mm, 7×7 mm and 5×5 mm, and different depths: 0.63 mm, 1.46 mm and 2.08 mm. Theoretically, the 12×12 mm defect at 0.63 mm depth is the easiest to detect and the 5×5 mm at 2.08 mm depth is the most difficult.

To collect the data, 36 inspections are carried out. Each inspection consists of a 20-second video with a Xenix Gobi 640 GigE infrared camera (see Table I for more information), in which the sheet is heated for 10 seconds. The camera is positioned 1.5 meters from the specimen and the two halogen lamps at 1.6 meters. Fig. 3 shows the temperature signal over time. The video runs at 50 FPS, therefore 1000 frames are obtained in each inspection. Between the inspections the sheet is cooled for 10 seconds. To achieve more variability in the data, the part is rotated 10° in each inspection, ensuring that the sheet is always kept on the same plane using a laser system to check the correct position.

To reduce the noise in the data, Principal Component Thermography (PCT), which is based on Principal Component

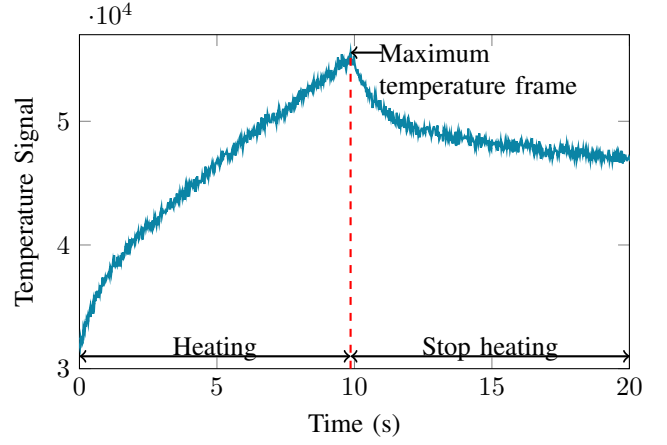


Fig. 3. Time of the inspection.

TABLE I
TECHNICAL SPECIFICATIONS OF THE INFRARED CAMERA XENIX GOBI 640 GIG E USED IN THE EXPERIMENTS.

Camera	Xenix Gobi 640 GigE
Temperature range	-20°C to 120°C
Spectral range	8–14 μm
Image frequency	50 Hz
Pixel resolution	640×480

Analysis (PCA), is used. PCT is used to reduce the number of variables without affecting the subsurface defect detection. PCT decomposes an image sequence into empirical orthogonal functions (EOF). EOF creates a set of orthogonal statistical modes that provide the best projection for the data [31]. The frames obtained during the inspection are represented on the left of Fig. 4. After applying PCT to the video, frames 1, 3 and 4 on the right of Fig. 4, are those selected to create an RGB image (see Fig. 5) since they are the ones that maximize the signal-to-noise ratio (SNR). The SNR metric is calculated using Eq. (1), where Def_u is the arithmetic mean of all the pixels inside the defect area, Ref_u is the arithmetic mean of all of the pixels that do not belong to a defect and Ref_σ is the standard deviation of all the pixels that do not belong to a defect [11]. These frames are selected because they have the greatest contrast between the defects and the background, as shown in Fig. 6. This technique of selecting 3 frames from the video has been used in other works [32], [33]. Considering that it is not possible to use all the non-defects pixels of the sheet as reference (Ref_u) to calculate the SNR, for each defect it has been selected an exclusion and a reference area [34]. Fig. 7 shows these areas. The area between the reference area and the exclusion area has been used as reference (Ref_u and Ref_σ). Table II shows the SNR for each defect before and after applying PCT. These results demonstrate that PCT is not only useful for summarizing thermographic data, but also for improving the SNR compared to the raw data. Selecting the frames that maximize the SNR is difficult, since it depends on the properties of the anomalies [11], therefore for new samples the frames that maximize the SNR would have to be analyzed.

$$SNR = 20 \log_{10} \left(\frac{|Def_u - Ref_u|}{Ref_\sigma} \right) \quad (1)$$

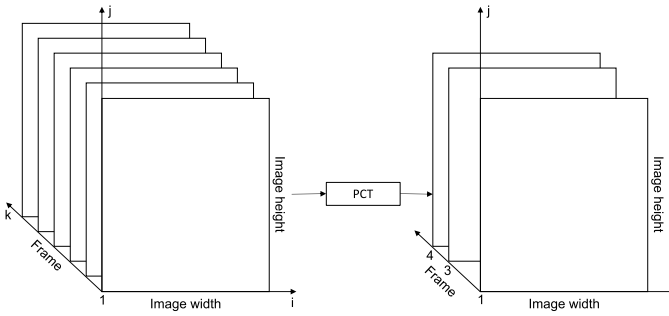


Fig. 4. Noise reduction obtained in the inspection, improving the contrast ratio between the defect and the background by applying PCT.

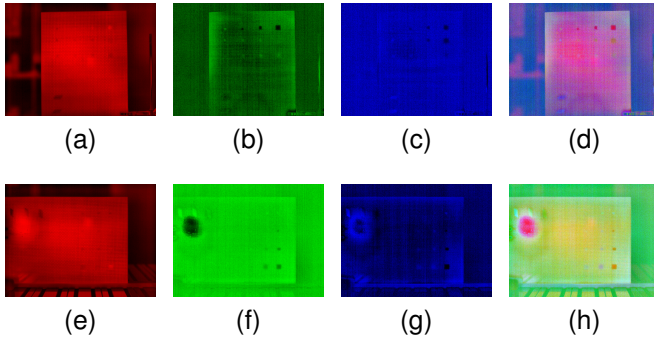


Fig. 5. Result of combining frames 1 (a, e), 3 (b, f) and 4 (c, g) into a RGB image (d, h).

Once the 36 3-channel images are collected, the next task is to label them. Labeling images containing subsurface defects can be an arduous process. To automate this process, the image obtained from the first inspection is carefully labeled manually (see Fig. 8a), following the known positions (see Fig. 2b). Before labeling the first inspection, the image is preprocessed with CLAHE (Contrast Limited Adaptive Histogram Equalization) [35]. This improves the contrast, as can be seen in Fig. 8a, the illumination can lead to confusion. This would make the ground truth incorrect. For the rest of the images, where the sheet is moved and rotated (see Fig. 8b), the edges are calculated using the Canny algorithm [36] (see Fig. 8c). Since corners detectors would generate many false positives

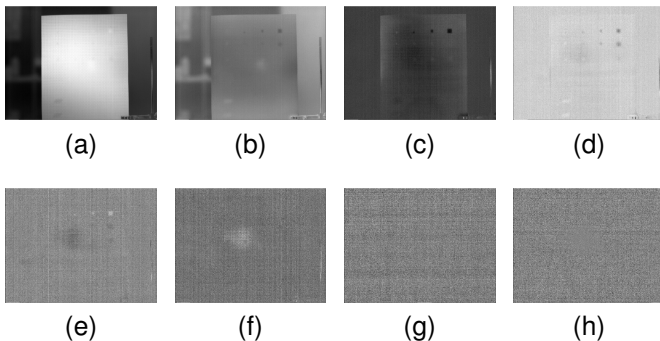


Fig. 6. Result of applying PCT to the video. (a) Frame 1. (b) Frame 2. (c) Frame 3. (d) Frame 4. (e) Frame 5. (f) Frame 6. (g) Frame 262. (h) Frame 494.

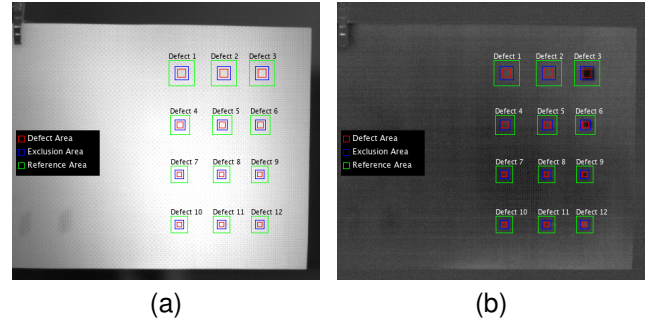


Fig. 7. Defect, exclusion and reference areas of (a) maximum temperature frame before PCT and (b) third frame after PCT.

TABLE II
SNR FOR EACH OF THE DEFECTS BEFORE APPLYING PCT ON THE MAXIMUM HEATING FRAME, AND AFTER APPLYING PCT ON THE THIRD FRAME.

	Before PCT	After PCT
Defect 1	-3.57	-1.28
Defect 2	-18.10	-5.73
Defect 3	1.31	19.45
Defect 4	-8.23	-1.60
Defect 5	-14.85	-9.59
Defect 6	-8.72	11.71
Defect 7	-31.57	-7.81
Defect 8	-14.50	-17.15
Defect 9	-12.96	9.19
Defect 10	-7.70	-22.48
Defect 11	-24.82	-5.20
Defect 12	-3.73	3.79

(they would also detect the corners of the support), the Hough transformation is applied to search for lines on the binary edge image [37]. In this way, the corners are calculated as the intersection of the lines (see Fig. 8d). Next, the corner correspondences between the base image and the rotated image are calculated (see Fig. 8e). From these correspondences, the 2D rigid transformation, calculated as shown in Eq. (2) where x and x' are the homogeneous coordinates of the point at the original and transformed position respectively, R is the rotation matrix and t is the translation vector. The translation vector determines the movement and rotation of the sheet [38]. This transformation relates the rotation and translation of the base image corners and those of the rotated image. Through the application of this transformation, the corners of the subsurface defects bounding boxes are calculated (see Fig. 8f). By applying this method to all the images, the dataset is automatically labeled, which saves time and ensures that the labeling is correct.

$$x' = Hx = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} x \quad (2)$$

Due to the limited number of images, after dividing the dataset into a 12-image test set and a 24-image training set, the augmentation is applied to the training set. The augmentation consists of the following modifications: $\pm 5^\circ$ rotation, vertical and horizontal flips, 1% to 20% zoom variations and random elastic image distortion. After applying this augmentation technique, 500 images for training are obtained. Fig. 9 shows

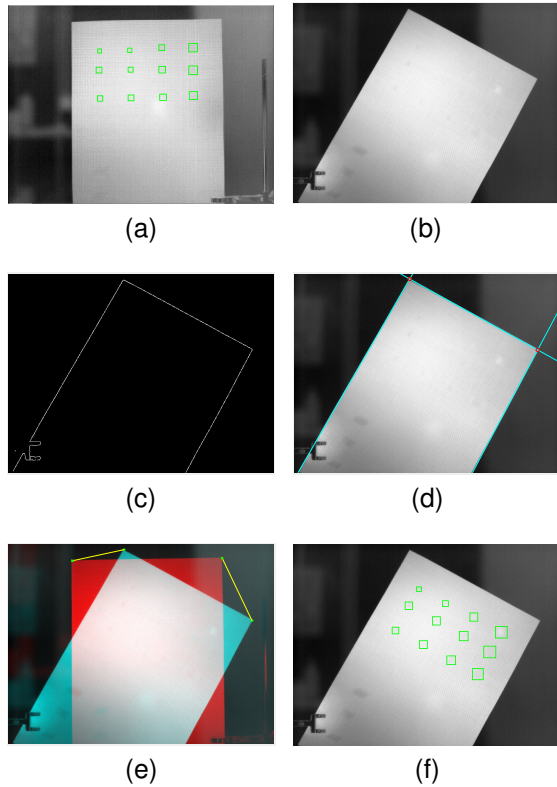


Fig. 8. Autolabelling process. (a) Base image with its ground truth. (b) Rotated image. (c) Edge calculation. (d) Hough Transformation. (e) Corner calculation from applying 2D rigid transformation. (f) Rotated image with its ground truth.

the result of mixing frames 1, 3 and 4, obtained after post-processing the inspection videos with PCT, into an RGB image and applying this augmentation technique to those corresponding to the training set. Some circular patterns can be seen in some of the images in Fig. 9. This is a common phenomenon in active inspections with optical thermography, produced by the reflection of the heating lamps on the surface of the inspected object. These reflections can hide other indications of interest. They can usually be avoided by modifying the inspection setup, although this is not always possible due to space limitations. Therefore, in order to reflect the conditions of a worst-case scenario, the effect of the presence of parasitic reflections in the results of the study, these reflections have not been avoided and can be seen in some of the images where the specimen is rotated with respect to the initial vertical position.

B. Analysis of classification algorithms

Image classification algorithms have helped automate tasks such as facial recognition or vehicle license plate recognition. To carry out these tasks, it is necessary to train one of these algorithms. There are several algorithms that have shown good performance, but due to the impossibility of evaluating all of them, two of those that offer the best results have been selected:

a) Efficientnet v2: This classifier [39] emerges as the natural evolution of its predecessor, Efficientnet [40]. In this work, it was observed that scaling all model dimensions

(depth, width and resolution) can improve the image classification results. For doing this, a new method is proposed. This method consists of scaling model dimensions uniformly with constant ratio. This innovation produces a bottleneck, when the resolution is too large and too many layers are used. This results in high training times. In addition, there is a point where increasing the resolution and the number of layers does not improve the results. The second version of this network, fix this problem by restricting the maximum resolution to 480, and by adding more layers to later stages.

b) Mobilenet v2: Mobilenet [41] is another widely used classifier. The core basis of this network is depthwise separable convolution. This is basically a depthwise convolution followed by a pointwise convolution. A depthwise convolution does the convolution individually for each channel, and then stack the three outputs (for RGB images). The next step is a pointwise convolution. This consists of applying a convolution to the new image generated in the previous step, with a $1 \times 1 \times 3$ kernel. This process is repeated 10 times, to generate a $W \times H \times 10$ feature map. Thanks to these steps, the computational costs are reduced by 7. The second version of this network [42], continues the essence of its predecessor. It uses an inverted residual structure where the input and output of the residual blocks are thin bottleneck layers.

C. Analysis of semantic segmentation algorithms

In semantic segmentation networks, there have also been great advances in recent years. Of all the networks developed, U-Net [43] is the most popular. Its name is due to its encoder-decoder architecture, which gives it a U-shape. The encoder is a stack of convolutional and max pooling layers. It acts as the feature extractor and learns through a sequence of the encoder blocks. Each block consists of two 3×3 convolution layers and a Relu activation function. The bridge connects the encoder and the decoder. It consists of two 3×3 convolutions. Finally, the decoder makes it possible to locate the objects of interest. It consists of 2×2 transpose convolution, concatenated with the corresponding skip connection from the encoder. The skip connections provide additional information that help the decoder generate better semantic features. Its popularity is due to its ability to produce good results using a small number of training images, and its ease of adapting the number of classes and input size.

D. Analysis of object detection algorithms

In recent years different object detection algorithms have been developed offering good results in datasets such as Pascal VOC [44] and COCO [45]. Object detectors can be divided into two types: one-state and two-state detectors. Two-state detectors first propose a set of regions of interest (ROIs) and then perform the appropriate detections in these regions. R-CNN [46] and its later versions, Fast R-CNN [47] and Faster R-CNN [48], are clear examples of two-state detectors. One-state detectors perform detection by treating the entire image. SSD [27] and YOLO [49] are the two most used one-state detectors. Typically, in datasets used to compare different detectors, two-state detectors achieve better results, at the cost

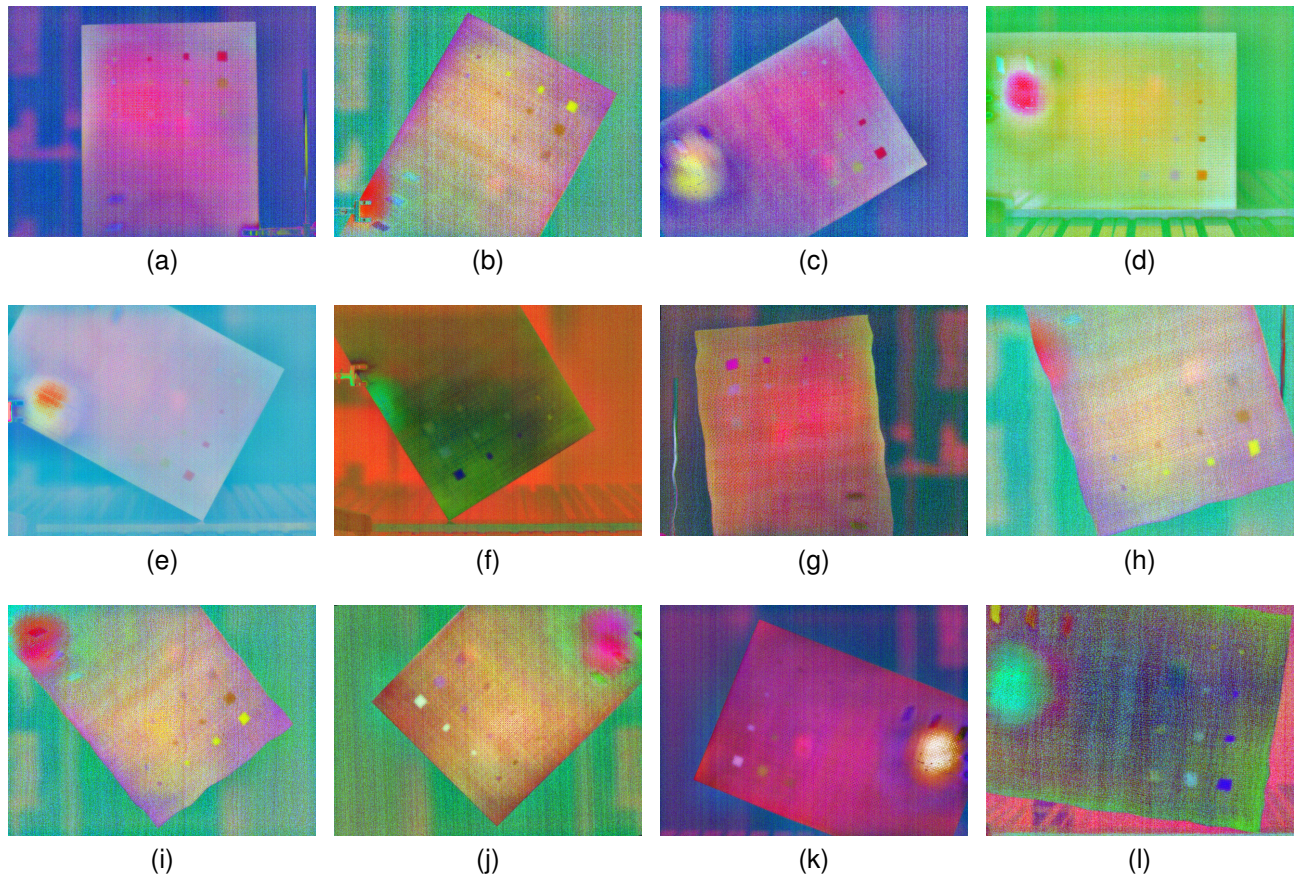


Fig. 9. Results after applying PCT and combining frames 1, 3 and 4 into a RGB image. (a–f) Test images. (g–l) Training images after applying augmentation.

of slower inference speed. Since the objective of this work is to be able to carry out the detection of defects in industrial parts by active thermography, the speed of inference is of vital importance. Thus, SSD and YOLO have been selected. These detectors and their many version are described below.

a) SSD: Single Shot Detector (SSD) [27] is a detector that competes in terms of accuracy with R-CNN and improves its inference time. SSD introduces a series of improvements, such as multi-scale detection, default boxes and aspect ratios, that enhance the detections. During training, the network uses hard negative mining, which ensures a proportion between the positives and negatives predictions of 1:3.

b) YOLOv1: You Only Look Once or YOLO [49] is the first version of this family of detectors. The main advantage introduced in YOLOv1 is inference in real time. YOLO divides the image into an $S \times S$ grid, where each resulting cell detects the objects whose center lies in that cell. As many redundant predictions are generated, non-maximal suppression (NMS) is applied to eliminate them.

Despite the many advantages introduced in YOLOv1, it has inferior results to other detectors such as R-CNN.

c) YOLOv2: The second version of YOLO [50] introduces a series of improvements, such as anchor boxes, to overcome some of the drawbacks of YOLOv1. In YOLOv1 it was necessary to make predictions from scratch, but this is not the case in YOLOv2. In any dataset, many of the objects often have the same shapes and aspect ratios. The anchor

boxes of YOLOv2 are initial guesses which take advantage of this, therefore the predictions are not generated from scratch, as in YOLOv1. The peculiarity of YOLOv2 lies in the way these anchor boxes are selected. In other works, such as [48], they are selected by hand. In YOLOv2 they are selected using k-means, which translates into better initial guesses, and therefore better results.

d) YOLOv3: One of the main problems of YOLOv2 is in the detection of small objects. YOLOv3 [28] addresses this problem. The main contribution of YOLOv3 is the ability to perform three-scale predictions. YOLOv3 uses Darknet-53 instead of Darknet-19, used in YOLOv2. The tradeoff for the improving in identifying small objects is an increase in inference speed.

e) YOLOv4 and YOLOv5: YOLOv4 [29] and YOLOv5 [30] are the natural continuations of YOLOv3, by different authors. Like other state-of-the-art detectors, both are composed of three parts: CSPDarknet-53 as the backbone, PA-NET as the neck and YOLOv3 as the head. By using cross-state partial networks (CSP) [51] YOLOv5 solves the problem of repeated information in the gradient, which results in a faster and smaller model. With path aggregation pyramid network (PANet) [52] as the neck, YOLOv5 improves the propagation of low-level features. Finally, the head of YOLOv5, based on YOLOv3, generates three predictions of different sizes.

In both detectors, several ideas are introduced that significantly improve the results of YOLOv3. Many of these ideas

are related to data augmentation. One of the best performing techniques is mosaic augmentation, which consists of creating an image from parts of other images. In this way, the model is trained with very diverse images and can therefore generalize correctly. Another idea introduced by YOLOv5 is the use of what are known as auto-anchor boxes. It continues to use k-means for the calculation of anchor boxes, but unlike previous versions, no configuration to use the anchor boxes that best fit the desired dataset is needed.

E. Evaluation metrics

The metrics used in image classification, object detection and semantic segmentation are related, but have some differences. A prediction can be computed as a True Positive (TP) in case it is correctly predicted, False Positive (FP) if it is incorrectly predicted, False Negative (FN) if it is incorrectly unpredicted or True Negative (TN) if it is correctly unpredicted. In image classification, it is just necessary to compare the predicted class with the expected one. In semantic segmentation, each pixel of the prediction is compared with its corresponding pixel of the ground truth, but in object detection, this direct comparison, is not possible. This is because the prediction and the ground truth do not normally match at 100%, therefore the intersection over the union (IOU) is used. The IOU, which measures the degree of overlap between two regions (D and G), is shown in Eq. (3). If the IOU between the prediction and the ground truth is greater than a threshold, typically 0.5, the prediction is considered as TP, otherwise as FP. The prediction is considered as an FN when an object present in the ground truth is not detected. In case of object detection, it is not possible to compute True Negatives (TN) as in every image it would be infinite.

With these basic metrics, it is possible to calculate the precision, recall and F_1 . Precision measures the percentage of correctness of the predictions made. Mathematically, precision is calculated as shown in Eq. (4). Recall, shown in Eq. (5), measures the percentage of predictions that have been correctly classified or detected. To weight precision and recall with a single value, the F_1 shown in Eq. (6) is used. These are the most popular metrics in image classification and semantic segmentation, but not in object detection. In this field, the most common metric used to evaluate the performance is the average precision (AP) [44]. This is because each prediction is given with a certain confidence value. Depending on the selected confidence threshold, the precision and the recall can vary. Each prediction is composed of a class name, a confidence score value and a bounding box. For computing the precision–recall curve, the predictions are sorted in ascending order by their confidence value. Fig. 10 shows the precision–recall curve obtained in this work with YOLOv5. The mean average precision (mAP), or AP as there is only one class, is the area under this curve, and it is non-dependent of the confidence value.

$$IOU = \frac{|D \cap G|}{|D \cup G|} \quad (3)$$

$$Precision = \frac{TPs}{TPs + FPs} \quad (4)$$

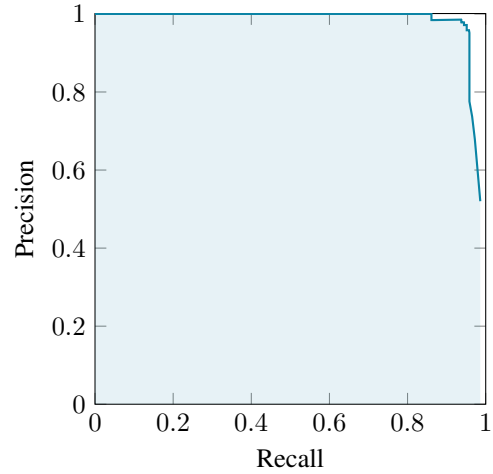


Fig. 10. Precision–recall curve of YOLOv5. The mAP is the area under the curve.

$$Recall = \frac{TPs}{TPs + FNs} \quad (5)$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (6)$$

In literature, when the dataset is composed of more than one class, it is common to use the mAP. The mAP, shown in Eq. (7), is the average of the APs for each of the classes. In this work since just one class is used, the mAP is not used. The AP can be calculated in several ways [53]:

- AP_{50} and AP_{75} are APs calculated with fixed IOU threshold of 0.5 and 0.75 respectively, for considering a prediction as a TP or FP.
- AP or AP@[.5:.05:.95] are the APs calculated as the average of the AP obtained for different IOU thresholds, from 0.5 to 0.95 with steps of 0.05.
- AP Across Scales is the AP calculated as in AP@[.5:.05:.95] but taking into account the size of the prediction. If the $bb_{x_{area}} < 32^2$ pixels, it is called AP_S . If $32^2 < bb_{x_{area}} < 96^2$, it is called AP_M . If $bb_{x_{area}} > 96^2$, it is called AP_L .

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (7)$$

III. RESULTS AND DISCUSSION

In this section, the results of the image classifiers, object detectors and image segmentation algorithms used are analyzed. All the experiments have been carried out with the following hardware: Intel Core i7 9700K CPU, 64GB of RAM and a NVIDIA GeForce RTX 2080 Ti Turbo GPU.

A. Classification results

The idea of evaluating image classification arises from the possible need to simplify the detection of subsurface defects. In this scenario, the dataset described in subsection II-A cannot be used. It was necessary to create an alternative one, in

TABLE III
RESULTS OBTAINED WITH EFFICIENTNET V2 AND MOBILENET V2.

	Precision	Recall	F_1
Efficientnet v2	0.8125	0.8636	0.8057
Mobilenet v2	0.8750	0.8810	0.8745

TABLE IV
RESULTS OBTAINED WITH U-NET.

	Precision	Recall	F_1
U-Net	0.689	0.717	0.703

which non-defects images were introduced. To compare it with semantic segmentation and object detection algorithms under the same circumstances, the same number of images for training and testing were used. For training the same data augmentation policy was also applied. The results shown in Table III, prove that it is possible to create a classifier for this kind of problems. Mobilenet v2 offers the best results, since its F_1 is a 7% higher than that of Efficientnet v2. These results were achieved with the following configuration: 50 epochs, a batch size of 16, a learning rate of 0.005, SGDM as solver and a input size of 224.

B. Semantic segmentation results

To compare semantic segmentation and object detection, U-Net is evaluated with the dataset described in subsection II-A, after generating the appropriate masks from the object detection labeling already made. After carrying out several experiments, the best hyperparameter setup is the following: 1000 epochs, batch size of 8, learning rate of 0.001 and solver Adam. With this configuration, the obtained results are shown in Table IV. These results prove that it is possible to use U-Net with this dataset, but the incorrectly predicted pixels cause metrics (precision and recall) to be lower than 0.75. For this reason, object detection is evaluated more thoroughly. Fig. 11 shows some predictions obtained with U-Net.

C. Object detection results

The following object detector algorithms to detect subsurface defects have been evaluated under different conditions: SSD, YOLOv3, YOLOv4 and YOLOv5.

Firstly, the dataset described in subsection II-A is used. The results obtained are summarized in Table V. In terms of AP, YOLOv5 outperforms the rest. It is 17% better than SSD, and 2% better than YOLOv3 and YOLOv4. For precision, recall and F_1 it is necessary to set a fixed value of confidence. To compare these metrics, the confidence threshold that maximizes the F_1 is established. By using this criterion, rather than a fixed confidence for all models, the best possible model for each object detector is compared. Thus, YOLOv5 achieves the best results with precision, recall and F_1 . The results shown in Table IV and Table V demonstrate that for this problem, object detection is a more suitable option than semantic segmentation.

As mentioned above, precision, recall and F_1 vary depending on the confidence threshold chosen. To measure the sensitivity of YOLOv5, Fig. 12 shows how these metrics vary

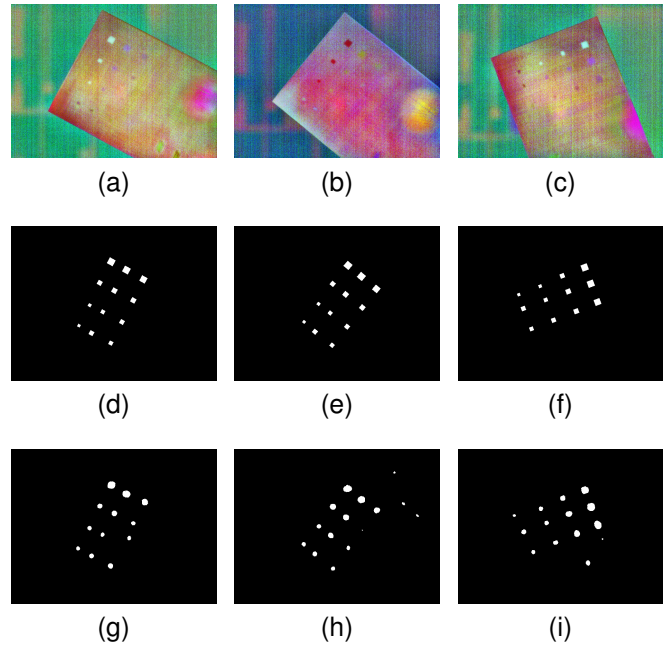


Fig. 11. Segmentation made with U-Net over test images. (a–d) Images. (e–h) Ground truth. (i–l) Predictions with U-Net.

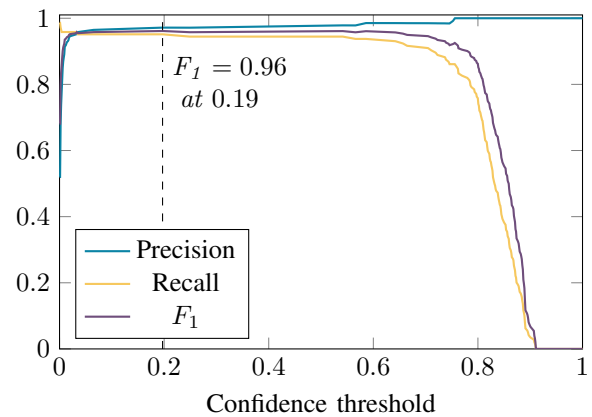


Fig. 12. Precision, recall and F1 curves for different confidence thresholds with YOLOv5.

in function of the confidence threshold. Although a confidence of 0.197 maximizes F_1 , the metrics are above 0.95 if the confidence threshold lies in the interval $[0.019, 0.663]$, which ensures the robustness of the object detector.

Fig. 13 shows a visual sample of the detections performed on some images of the test set. Fig. 13(a–d) show the ground truth, and Fig. 13(e–h), show the detections performed with SSD. SSD detects some of the defects but cannot locate them correctly. Fig. 13(i–l) show the detections carried out with YOLOv3. Fig. 13(j,k) show that YOLOv3 fails to detect many of the defects in the test set. In Fig. 13(m–p) the detections performed with YOLOv4 are shown. In this case, the results obtained with SSD and YOLOv3 are improved, however, Fig. 13o shows the detection of an FP. Finally, Fig. 13(q–t) show the detections performed with YOLOv5. Visually, YOLOv5 achieves better results than the other object detector

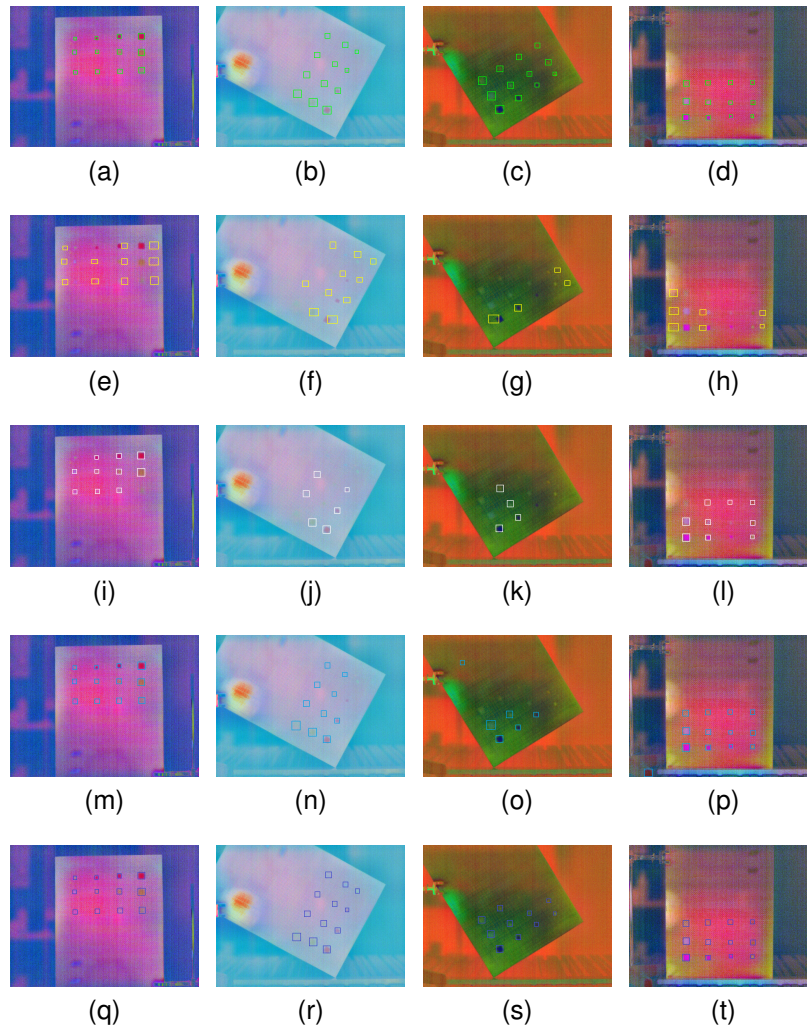


Fig. 13. Comparison of detections performed on a subset of the test. (a–d) Ground truth. (e–h) Detections performed with SSD. (i–l) Detections performed with YOLOv3. (m–p) Detections performed with YOLOv4. (q–t) Detections performed with YOLOv5.

algorithms, since it detects and locates most of the defects correctly, which confirms the metrics shown in Table V. The information obtained from applying these CNN-based object detectors to the PCT images can be extracted and applied to the original images. Since YOLOv5 offers the best results, Fig. 14 shows the detections performed with YOLOv5 on additional original test images before applying PCT.

To achieve these results several experiments are carried out with all the object detectors. After fine-tuning the hyperparameters, the best configurations for each detector are shown in Table VI. With all the detectors the original network augmentation is used. With these configurations, the training times are the following: SSD–Pytorch 5 hours and 57 minutes, YOLOv3–Pytorch 27 minutes, YOLOv4–Pytorch 28 minutes and YOLOv5–Pytorch 8 minutes. The whole YOLO family networks train quickly, therefore it would be possible to train a model with extra images from other parts. In the case of SSD the training time is around 6 hours. Although SSD is not as fast as the YOLOs, a model can be trained with SSD with several more parts, as long as the server can be dedicated to this task.

TABLE V
RESULTS OBTAINED WITH SSD, YOLOV3, YOLOV4 AND YOLOV5.

	Precision	Recall	F_1	Confidence	AP_{50}	AP
SSD	0.8105	0.8241	0.8172	0.311	0.8039	0.3267
YOLOv3	0.8977	0.9150	0.9063	0.101	0.9260	0.4106
YOLOv4	0.9687	0.8611	0.9117	0.271	0.9254	0.3967
YOLOv5	0.9716	0.9513	0.9614	0.197	0.9767	0.4831

TABLE VI
HYPERPARAMETERS USED DURING TRAINING.

	Epochs	Batch size	Learning rate	Backbone	Solver
SSD	15	16	0.01	Resnet50	sgdm
YOLOv3	100	8	0.001	Darknet53	adam
YOLOv4	100	8	0.01	CSPDarknet53	sgdm
YOLOv5	1000	8	0.001	CSPDarknet53	adam

Further experiments were carried out varying the algorithms configuration. Using more than 100 epochs produces overfitting, as there is not enough data and the model begins to memorize the training set. Choosing the optimum learning rate is also of vital importance, as otherwise, the model will

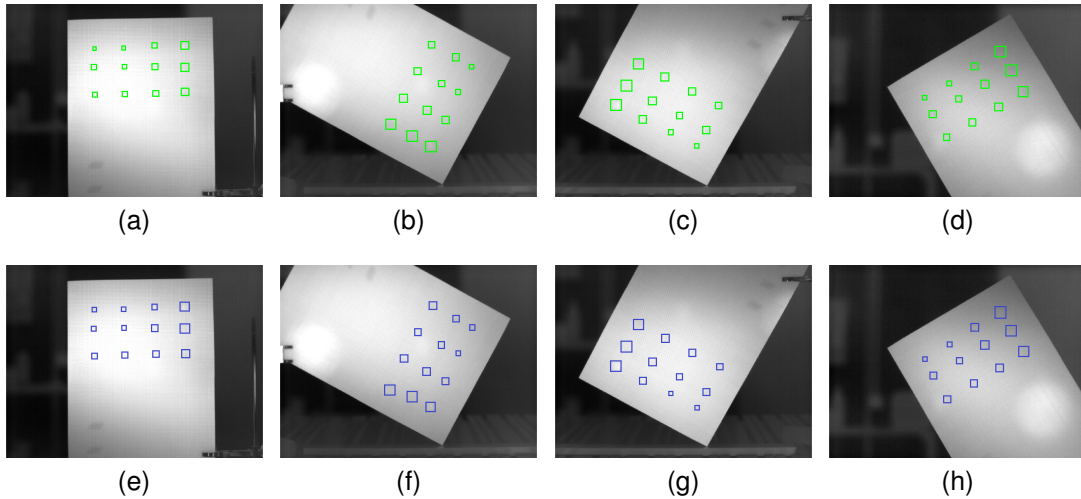


Fig. 14. Detections made with YOLOv5 over original test images (before applying PCT). (a–d) Ground truth. (e–h) Detections performed with YOLOv5.

not converge. In the experiments performed, the best learning rate is always in the interval $[0.01, 0.001]$. The learning rates outside this interval result in poor precision, recall and AP. Apart from the augmentation applied to generate more images to train the algorithms, the networks can use their own augmentation during training. This helps the model to generalize better, as the network modifies the training images in every epoch. To understand how this augmentation can affect the metrics behavior, experiments without it were done, obtaining an AP_{50} between 20% and 30% lower than with the network augmentation. Fig. 15 shows the loss function over the epochs in two experiments. The best experiment performed with YOLOv5 is shown in Fig. 15a. As the training and validation losses cross at a value close to zero, the experiment converges. Since the intersection point is around epoch 80, it is not necessary to use more epochs for training. Adding extra epochs would only produce overfitting, as the training loss would still decrease but not the validation loss. The same experiment can be seen in Fig. 15b, but without applying the network augmentation. The training loss function is lower than when augmentation is applied (Fig. 15a). This is because there is less variability in the data, therefore the network memorizes the training images. For this reason the validation loss function does not stop growing and the experiment does not converge. An mAP of 0.656 is obtained, 32% lower than if augmentation were used.

As mentioned above, due to the limited amount of data available, augmentation (not network augmentation) is applied to the training set in order to generate more images. The results of the experiments to evaluate the feasibility of training without applying augmentation are summarized in Table VII. The only difference between the experiments with and without augmentation, is that in the second case 500 epochs were used. These results confirm the need to apply data augmentation in order to generate more diversity in the training set.

These deep learning object detectors have also been evaluated with another division, in which of the 36 images, 18 are used for testing and 18 for training. In order to get more

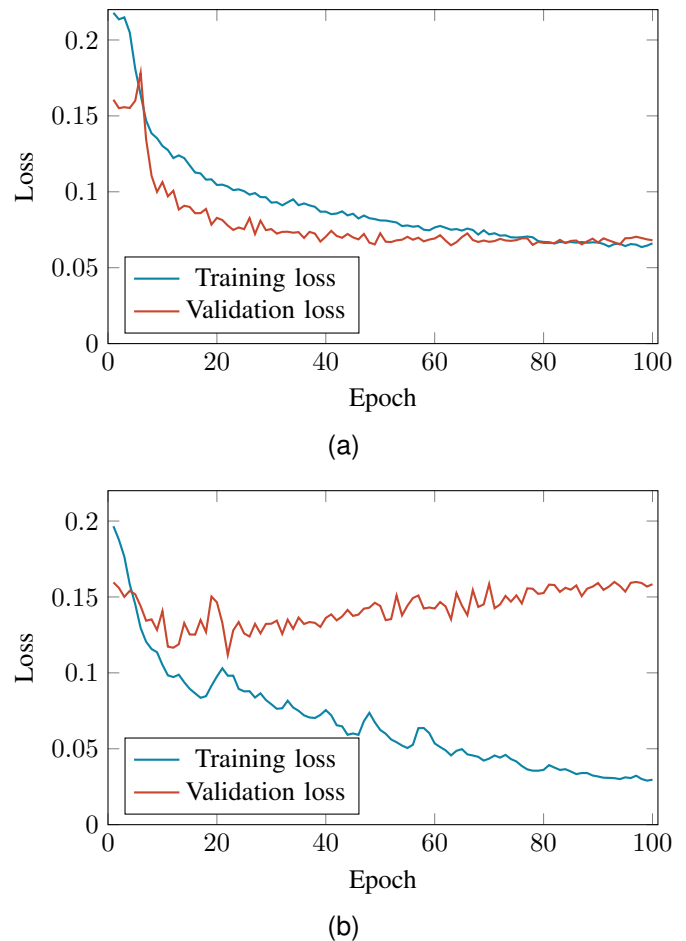


Fig. 15. Evolution of the loss function. (a) Best experiment obtained with YOLOv5. (b) Experiment without network augmentation with YOLOv5.

TABLE VII
RESULTS OBTAINED WITH SSD, YOLOv3, YOLOv4 AND YOLOv5
WITHOUT APPLYING AUGMENTATION TO THE TRAINING SET.

	Precision	Recall	F_1	Confidence	AP_{50}	AP
SSD	0.6921	0.3824	0.4926	0.157	0.4103	0.1322
YOLOv3	0.7471	0.4719	0.5785	0.167	0.4677	0.1662
YOLOv4	0.7303	0.5833	0.6486	0.237	0.5531	0.1741
YOLOv5	0.8031	0.6805	0.7368	0.230	0.6808	0.2303

TABLE VIII
RESULTS OBTAINED WITH SSD, YOLOv3, YOLOv4 AND YOLOv5 WITH
THE SECOND DATASET.

	Precision	Recall	F_1	Confidence	AP_{50}	AP
SSD	0.7721	0.7548	0.7633	0.424	0.7106	0.3166
YOLOv3	0.9621	0.8287	0.8904	0.219	0.8714	0.3848
YOLOv4	0.9707	0.9207	0.9450	0.181	0.9434	0.4154
YOLOv5	0.9858	0.9675	0.9766	0.141	0.9774	0.5178

images for training, the same augmentation is applied to the training set. The results are shown in Table VIII. In this case, where 18 images are used for training and a 18 for evaluation, the results are very similar to those of the original dataset, where 24 images are used for training and 12 for evaluation. For this reason, it is concluded that both options are suitable to detect subsurface defects in industrial parts, as long as data augmentation is applied.

In industry not only is the accuracy of the detections important, but also the inference speed. Fig. 16 shows the inference speed of each of the object detector algorithms evaluated in this work, using a 640×480 image. The inference times, with a GeForce RTX 2080 Ti Turbo GPU, are the following: SSD – 8 ms, YOLOv3 – 12 ms, YOLOv4 – 14 ms and YOLOv5 – 6 ms. These are the average times of running the inference over 10 images. YOLOv5 is seen to offer the best performance, not only for precision, recall and AP_{50} , but also in inference time. All of these detectors can run in real time, which speeds up the process of evaluating an industrial part.

Since one of the objectives is to be able to generalize, the resulting model with YOLOv5 is used to detect subsurface defects in sheets with different but similar materials. The alternative materials chosen are a foam sandwich panel and a

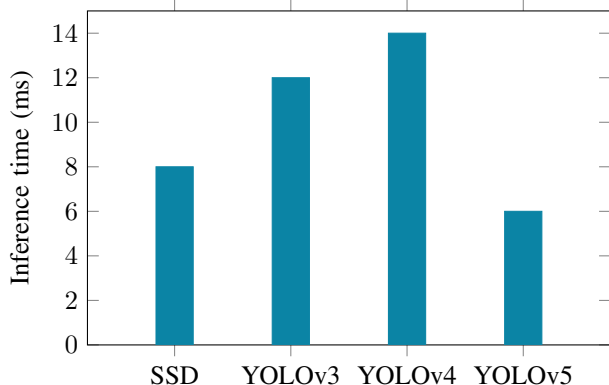


Fig. 16. Time to process a specimen using SSD, YOLOv3, YOLOv4 and YOLOv5.

honeycomb sandwich panel. These panels are selected because their use is widespread in various industries, such as aerospace, building and shipbuilding. Their properties include thermal insulation, lightness, strength and stiffness. Early detection of subsurface defects is of vital importance as it can prevent construction failures or damage caused during operations that could lead to high repair costs and long downtimes. Fig. 17 shows the two parts used, each one with a different type of core. Each part has 12 known subsurface defects, but to make the detection more challenging, several parts are used in which the depth of the defects is varied. That is, the model is evaluated with defects at different depths from those used for training. Fig. 18 shows the results of these detections. Fig. 18a is the same material as that used for training but with the defects at different depths. In this case, all the defects are detected correctly. Fig. 18b and Fig. 18c are two foam sandwich panels, Fig. 18b with defects at the same depth as the material used for training and Fig. 18c with defects at different depths. When using a foam sandwich panel with subsurface defects at the same depth as those of the material used for training, the recall is greater than when using a foam part with subsurface at different depths. Finally, Fig. 18d and Fig. 18e are two honeycomb sandwich panels, Fig. 18d with defects at the same depth as the material used for training and Fig. 18e with defects at different depths. It is plain to see that detection with a honeycomb sandwich panel is more feasible than with a foam sandwich panel. The results obtained with YOLOv5 prove that it is possible to train a model with one material and use it with different materials, but ideally to achieve optimum performance, a dataset with all materials to be inspected should be created and used for training.

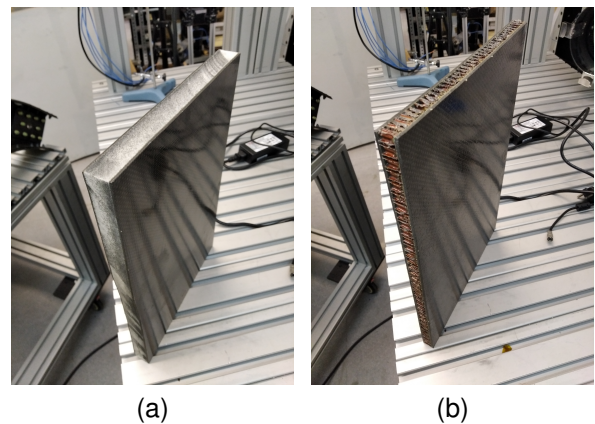


Fig. 17. Alternative specimens used to evaluate the model generalization. (a) Foam sandwich panel. (b) Honeycomb sandwich panel.

IV. CONCLUSION

In this work, active thermography (IRT) in combination with deep learning is proposed to detect subsurface defects. With deep learning it is possible to inspect an industrial part objectively. To collect a dataset, 36 20-second infrared videos were made. In each of the 36 inspections the sheet is heated for 10 seconds and cooled another 10 seconds. Principal Components Thermography (PCT) is used as a post-processing

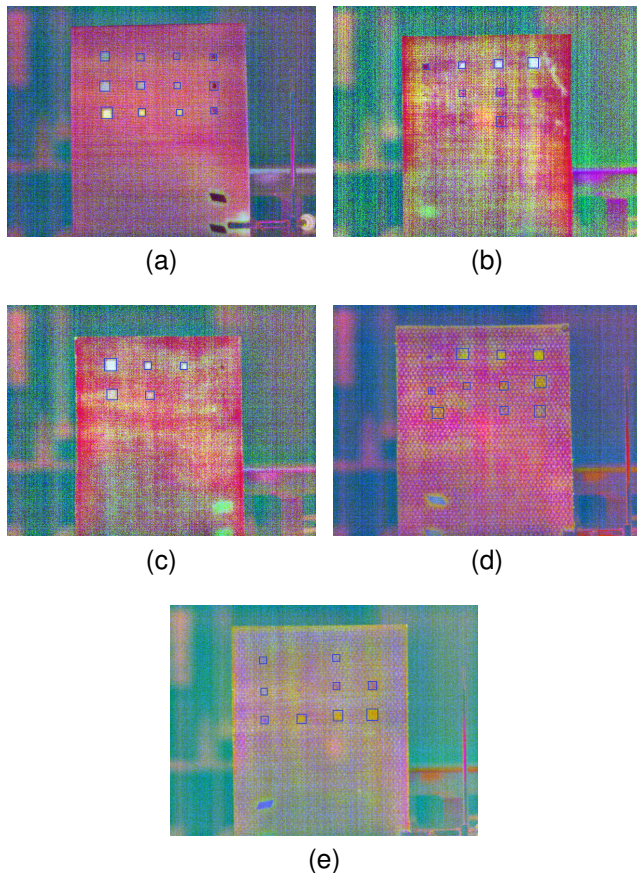


Fig. 18. Detections performed with YOLOv5 on sheets not used during training. (a) Same material structure as that used during training but with different subsurface defect depths. (b) Foam sandwich panel with defects at the same depths as the sheet used for training. (c) Foam sandwich panel with defects at different depths from the sheet used for training. (d) Honeycomb sandwich panel with defects at the same depths as the sheet used for training. (e) Honeycomb sandwich panel with defects at different depths from the sheet used for training.

technique since it has proved to improve the signal-to-noise ratio (SNR). As a result of applying PCT to each video, frames 1, 3 and 4 were selected to create an RGB image. Since labeling can be an arduous task an auto-labeling framework is created. Due to the limited amount of data, an augmentation technique is applied to the training set to generate 500 images. With these dataset image classification, semantic segmentation and object detection were evaluated. Image classification has proved good potential for resolving this kind of problems, but the impossibility of locating defects makes its less relevant than semantic segmentation and object detection. With the 36-image dataset U-Net, SSD, YOLOv3, YOLOv4 and YOLOv5 were evaluated under different conditions. The results obtained with U-Net demonstrate that although it is possible to use semantic segmentation, it is not the best option since there are some outliers pixels that reduce the performance metrics. For this reason, object detection has been more widely explored. The average precision (AP_{50}) obtained with each detector is of 0.8039, 0.9260, 0.9254 and 0.9767, respectively. In industry not only is detecting the defects important, but also the time needed to detect them. In terms of speed YOLOv5

also overcomes the other object detector algorithms, therefore parts can be inspected in real time. These results demonstrate that it is not necessary to develop a customized deep learning network to detect subsurface defects.

This work also shows the feasibility of training a model with a single carbon fiber solid laminate sheet, and using it to inspect sheets of other materials, although it would be ideal to train the model with sheets of all the different materials to be inspected.

The use of deep learning and IRT not only improves the quality control of industrial parts, but also reduces its cost. For future work, it would be useful to create a dataset with several industrial sheets composed of different materials, including internal and external defects, to combine the defect detection process in a single deep learning model.

ACKNOWLEDGMENTS

This work has been partially funded by the project RTI2018-094849-B-I00 of the Spanish National Plan for Research, Development and Innovation.

V. REFERENCES

REFERENCES

- [1] Y. F. Shu, B. Li, X. Li, C. Xiong, S. Cao, and X. Y. Wen, "Deep learning-based fast recognition of commutator surface defects," *Measurement*, vol. 178, p. 109324, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0263224121003237>
- [2] Y. Zong, J. Liang, H. Wang, M. Ren, M. Zhang, W. Li, W. Lu, and M. Ye, "An intelligent and automated 3d surface defect detection system for quantitative 3d estimation and feature classification of material surface defects," *Optics and Lasers in Engineering*, vol. 144, p. 106633, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0143816621001032>
- [3] D. Li, Q. Xie, X. Gong, Z. Yu, J. Xu, Y. Sun, and J. Wang, "Automatic defect detection of metro tunnel surfaces using a vision-based inspection system," *Advanced Engineering Informatics*, vol. 47, p. 101206, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1474034620301750>
- [4] Y. He, K. Song, Q. Meng, and Y. Yan, "An end-to-end steel surface defect detection approach via fusing multiple hierarchical features," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 4, pp. 1493–1504, 2020.
- [5] M. W. Kozak, "Radiation protection and safety in industrial radiography, international atomic energy agency safety series number 13," 2000.
- [6] R. Gade and T. B. Moeslund, "Thermal cameras and applications: a survey," *Machine vision and applications*, vol. 25, no. 1, pp. 245–262, 2014.
- [7] R. Usamentiaga, M. Yacine, C. Ibarra-Castanedo, M. Klein, M. Genest, and X. Maldague, "Automated dynamic inspection using active infrared thermography," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5648–5657, 12 2018, jCR: 7.377 - Q1 [2018]. [Online]. Available: <https://doi.org/10.1109/TII.2018.2836363>
- [8] C. Maierhofer and M. Röllig, "Active thermography for the characterization of surfaces and interfaces of historic masonry structures," in *Proceedings of the 7th International Symposium on Non-Destructive Testing in Civil Engineering (NDTCE), Nantes, France*, vol. 30. Citeseer, 2009.
- [9] F. Sham, N. Chen, and L. Long, "Surface crack detection by flash thermography on concrete surface," *Insight-Non-Destructive Testing and Condition Monitoring*, vol. 50, no. 5, pp. 240–243, 2008.
- [10] B. Wiecek, "Review on thermal image processing for passive and active thermography," in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, 2005, pp. 686–689.
- [11] R. Usamentiaga, P. Venegas, J. Guerediaga, L. Vega, J. Molleda, and F. G. Bulnes, "Infrared thermography for temperature measurement and non-destructive testing," *Sensors*, vol. 14, no. 7, pp. 12305–12348, 2014. [Online]. Available: <https://www.mdpi.com/1424-8220/14/7/12305>

- [12] R. Usamentiaga, P. Venegas, J. Guerediaga, and L. Vega, "Towards automatic defect detection in carbon fiber composites using active thermography," *Quantitative InfraRed Thermography*, 2014.
- [13] R. Usamentiaga, C. Ibarra-Castanedo, M. Klein, X. Maldague, J. Peeters, and A. Sanchez-Beato, "Nondestructive evaluation of carbon fiber bicycle frames using infrared thermography," *Sensors*, vol. 17, no. 11, 2017. [Online]. Available: <https://www.mdpi.com/1424-8220/17/11/2679>
- [14] S. J. Mambou, P. Maresova, O. Krejcar, A. Selamat, and K. Kuca, "Breast cancer detection using infrared thermal imaging and a deep learning model," *Sensors*, vol. 18, no. 9, 2018. [Online]. Available: <https://www.mdpi.com/1424-8220/18/9/2799>
- [15] M. de Freitas Oliveira Baffa and L. Grassano Lattari, "Convolutional neural networks for static and dynamic breast infrared imaging classification," in *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, 2018, pp. 174–181.
- [16] S. Guan, N. Kamona, and M. Loew, "Segmentation of thermal breast images using convolutional and deconvolutional neural networks," in *2018 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, 2018, pp. 1–7.
- [17] A. Akula, A. K. Shah, and R. Ghosh, "Deep learning approach for human action recognition in infrared images," *Cognitive Systems Research*, vol. 50, pp. 146–154, 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1389041717302206>
- [18] T. Kawashima, Y. Kawanishi, I. Ide, H. Murase, D. Deguchi, T. Aizawa, and M. Kawade, "Action recognition from extremely low-resolution thermal image sequence," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2017, pp. 1–6.
- [19] B. Yousefi, D. Kalhor, R. Usamentiaga Fernández, L. Lei, C. I. Castanedo, X. P. Maldague *et al.*, "Application of deep learning in infrared non-destructive testing," *QIRT 2018 Proceedings*, 2018.
- [20] R. Marani, D. Palumbo, U. Galietti, and T. D'Orazio, "Deep learning for defect characterization in composite laminates inspected by step-heating thermography," *Optics and Lasers in Engineering*, vol. 145, p. 106679, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0143816621001494>
- [21] Q. Luo, B. Gao, W. Woo, and Y. Yang, "Temporal and spatial deep learning network for infrared thermal defect detection," *NDT & E International*, vol. 108, p. 102164, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0963869519301355>
- [22] Y. Dong, C. Xia, J. Yang, Y. Cao, Y. Cao, and X. Li, "Spatio-temporal 3d residual networks for simultaneous detection and depth estimation of cfrp subsurface defects in lock-in thermography," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2021.
- [23] R. Ali and Y.-J. Cha, "Subsurface damage detection of a steel bridge using deep learning and uncooled micro-bolometer," *Construction and Building Materials*, vol. 226, pp. 376–387, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0950061819319671>
- [24] R. Moreno, E. Gorostegui-Colinas, P. L. de Uralde, and A. Muniategui, "Towards automatic crack detection by deep learning and active thermography," in *Advances in Computational Intelligence*, I. Rojas, G. Joya, and A. Catala, Eds. Cham: Springer International Publishing, 2019, pp. 151–162.
- [25] D.-M. Tsai, S.-K. S. Fan, and Y.-H. Chou, "Auto-annotated deep segmentation for surface defect detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–10, 2021.
- [26] J. Dong, Q. Chen, S. Yan, and A. Yuille, "Towards unified object detection and semantic segmentation," in *European Conference on Computer Vision*. Springer, 2014, pp. 299–314.
- [27] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.
- [28] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
- [29] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.
- [30] G. Jocher, A. Stoken, J. Borovec, NanoCode012, ChristopherSTAN, L. Changyu, Laughing, tkianai, yxNONG, A. Hogan, lorenzomamma, AlexWang1900, A. Chaurasia, L. Diaconu, Marc, wanghaoyang0106, ml5ah, Doug, Durgesh, F. Ingham, Frederik, Guilhen, A. Colmagro, H. Ye, Jacobsolawetz, J. Poznanski, J. Fang, J. Kim, K. Doan, and L. Yu, "ultralytics/yolov5: v4.0 - nn.SiLU() activations, Weights & Biases logging, PyTorch Hub integration," January 2021. [Online]. Available: <https://doi.org/10.5281/zenodo.4418161>
- [31] N. Rajic, "Principal component thermography for flaw contrast enhancement and flaw depth characterisation in composite structures," *Composite Structures*, vol. 58, no. 4, pp. 521–528, 2002. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0263822302001617>
- [32] P. Venegas, R. Usamentiaga, J. Peran, and I. Saez, "Quaternion Processing Techniques for Color Synthesized NDT Thermography," *Applied Sciences*, vol. 11, no. 2, pp. 1–24, 2021, jCR: 2.217 - Q2 [2018]. [Online]. Available: <http://dx.doi.org/10.3390/app11020790>
- [33] —, "Advances in rgb projection technique for thermographic ndt: Channels selection criteria and visualization improvement," *International Journal of Thermophysics*, vol. 39, no. 8, pp. 1–29, 2018, jCR: 0.853 - Q4 [2018]. [Online]. Available: <https://doi.org/10.1007/s10765-018-2417-9>
- [34] R. Usamentiaga, C. Ibarra-Castanedo, and X. Maldague, "More than fifty shades of grey: Quantitative characterization of defects and interpretation using snr and cnr," *Journal of Nondestructive Evaluation*, vol. 37, no. 2, p. 25, Mar 2018. [Online]. Available: <https://doi.org/10.1007/s10921-018-0479-z>
- [35] A. M. Reza, "Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement," *Journal of VLSI signal processing systems for signal, image and video technology*, vol. 38, no. 1, pp. 35–44, Aug 2004. [Online]. Available: <https://doi.org/10.1023/B:VLSI.0000028532.53893.82>
- [36] W. Rong, Z. Li, W. Zhang, and L. Sun, "An improved canny edge detection algorithm," in *2014 IEEE International Conference on Mechatronics and Automation*, 2014, pp. 577–582.
- [37] R. Ji and L. Qi, "Crop-row detection algorithm based on random hough transformation," *Mathematical and Computer Modelling*, vol. 54, no. 3, pp. 1016–1020, 2011, mathematical and Computer Modeling in agriculture (CCTA 2010). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0895717710005212>
- [38] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.
- [39] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 10096–10106. [Online]. Available: <https://proceedings.mlr.press/v139/tan21a.html>
- [40] —, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 6105–6114. [Online]. Available: <https://proceedings.mlr.press/v97/tan19a.html>
- [41] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
- [42] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [43] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
- [44] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun 2010. [Online]. Available: <https://doi.org/10.1007/s11263-009-0275-4>
- [45] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.
- [46] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.
- [47] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.
- [48] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

- [49] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.
- [50] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.
- [51] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "Cspnet: A new backbone that can enhance learning capability of cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 390–391.
- [52] K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng, "Panet: Few-shot image semantic segmentation with prototype alignment," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9197–9206.
- [53] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics*, vol. 10, no. 3, 2021. [Online]. Available: <https://www.mdpi.com/2079-9292/10/3/279>

VI. BIOGRAPHY SECTION