RESEARCH ARTICLE

Improved detection of subsurface defects through active thermography and ensembling techniques

Darío G. Lema[a], Oscar D. Pedrayes[a], Rubén Usamentiaga[a] and Daniel F. García[a]

[a]Department of Computer Science and Engineering, University of Oviedo, Campus de Viesques, 33204 Gijón, Asturias, Spain

ARTICLE HISTORY
Compiled February 9, 2023

ABSTRACT
Quality control and defect detection are major challenges in industrial environments. The localization of these defects is of crucial importance, as they can affect the performance of manufactured products and even the safety of people. Defect detection methods based on visual sensors and image processing are nowadays the most common approaches. To detect subsurface defects it is common to combine active thermography and deep learning. In this paper, PCT (Principal Components Thermography) is employed to enhance the SNR (signal-to-noise ratio), leading to a significant improvement in the results. Recent developments in this field apply deep learning models for semantic segmentation and object detection. However, the quality of the predictions is not enough to ensure that all quality controls are met. In this paper, a combination of semantic segmentation and object detection is proposed to increase the reliability of predictions. To carry out this combination, a working methodology and two new ensembling strategies are proposed. After comparing the results of the proposed combination with the results of using only semantic segmentation methods in a real industrial scenario, where carbon fiber sheets are used, it is found that the proposal improves the segmentation metrics by a 5% to a 24%. Thus the reliability of the predictions is improved.

KEYWORDS
Quality control; Subsurface defect detection; Model ensembling; Semantic segmentation; Object detection

1. Introduction

Quality control is of vital importance in industry, which leads to continuous innovation in defect detection methods. Particular emphasis has been placed on identifying surface defects, but there is still room for improvement in the detection of subsurface defects. These defects are not visible to humans, so their proper localization represents a major challenge.

Non-destructive testing (NDT) is a set of techniques to evaluate materials or components without damaging them Inagaki, Ishii, and Iwamoto (1999). Among the set of NDT techniques there are several non-contact solutions for inspecting subsurface defects. Some of the most common solutions are X-ray and infrared thermography (IRT). X-rays, although more precise than thermography, are more hazardous to operators.

CONTACT D.G.L. Author. Email: gonzalezdario@uniovi.es

Infrared thermography, in addition to be contactless, does not have the damaging properties of X-rays. These two facts make IRT a suitable method for the inspection of industrial parts.

Thermography can be divided into two types: passive and active. In passive thermography, the temperature contrasts of a body is measured without adding an external heat source. In active thermography, an external heat source is added to maximize these contrasts Usamentiaga et al. (2013). Inert objects, as is the case of the sheets to be evaluated in this work, do not present temperature variations. For this reason, active thermography is used. The induced heat affects differently the areas with subsurface defects than the areas without defects. This way, it is possible to detect the defects.

Because the defects are subtle, in this work, after applying active thermography and capturing the images with infrared cameras, Principal Component Thermography (PCT) is applied as a post-processing technique to enhance the signal-to-noise ratio (SNR) of the data.

Until relatively recently, defect inspection was performed manually by a experienced worker Usamentiaga et al. (2017). This made the inspection time-consuming and subjective. However, in recent years, the trend of combining IRT with deep learning to automate the process has emerged.

Thanks to the popularization of GPUs (Graphics Processing Unit) it is possible to train deep learning models with large amounts of data. These models offer high accuracy with high inference speed.

The combination of deep learning-based algorithms with active thermography has been applied in various fields. In medicine, several works have been done in which the conditions of patients are improved. In Lyra et al. (2021), deep learning and active thermography are combined to monitor vital signs of Intensive Care Unit (ICU) patients. To do this, they create a model based on YOLOv4-Tiny in which thermographic images are used to monitor chest movements caused by breathing. As the active thermography is contactless, it is possible to create a system that reduces the need to use technology based on wired sensors. In Cruz-Vega et al. (2020), both technologies are combined to detect Diabetes Mellitus. To do this, they create a classifier that is fed by thermographic images. In this work, traditional image classification techniques are compared with deep learning based methods. After a series of experiments, it is shown that deep learning based methods give better results than the traditional ones.

Deep learning and IRT are also used in the food industry. In Izquierdo et al. (2020), to detect possible food frauds, honey is adulterated with concentrations of rice syrup. To detect these frauds, they use a thermographic camera to extract images of the honey. These images are then used together with models based on CNNs (Convolutional Neural Network) to identify possible alterations. The results confirm that this combination of technologies is a valid option for improving food quality control. In Estrada-Pérez et al. (2021), both technologies are combined to detect adulterated rice grains. After conducting several experiments, it is possible to detect adulterations with an accuracy of 98%.

Defect control in industrial sheets is another field where the combination of these technologies offers great results. In Bang, Park, and Jeon (2020), both techniques are combined to detect defects in composite materials. For this purpose, a dataset has been developed from the available literature and a framework has been created in which the thermography process is integrated with a deep learning technique. In Fang, Ibarra-Castanedo, and Maldague (2021), the possibility of creating synthetic thermographic data to create robust deep learning models has been explored. In Yousefi et al. (2018),

an image classification technique for detecting defects using thermography is tested. In Lema et al. (2022), active thermography for detecting subsurface defects is combined with deep learning. For this purpose, a proprietary dataset is created and a large number of experiments are performed. In Liu et al. (2022), to locale subsurface flaws in polymer composites a convolutional graphic thermography (CGT) method is proposed. The goal is to be able to use low quality thermal images. For this purpose, they create a convolutional thermogram enhancer with which the image noise is reduced. The features of the enhanced thermograms are then extracted, allowing defects to be visually highlighted.

In Lema et al. (2022), the problem of subsurface defect detection is solved using object detection solutions. In Pedrayes et al. (2022), the problem is solved using semantic segmentation techniques. The main difference between object detection and semantic segmentation is that, in object detection the objective is to locate the objects of interest by bounding boxes. In semantic segmentation, each image pixel is classified.

In both papers, great results are achieved by combining active thermography and deep learning. However, neither work attempts to improve the results obtained by model ensembling in carbon fiber sheets. Model ensemble is a process in which multiple models are combined to predict a result. After each base model generates a prediction, these are combined according to a specific criterion. The objective in mixing the results of several models is to reduce possible errors, since if the models are independent of each other, the prediction error will be higher. Various ensembling techniques are covered in the literature:

- Adaptive Boosting (AdaBoost): this technique is based on training models from the errors generated by a previous model Freund and Schapire (1997). The first model is trained with the whole training set. Then, another model is trained, using the incorrect predictions made by the previous model as its training set. The predictions of both models together are more robust than those of the first model. Theoretically, the more iterations performed, the more accurate the final model will be. Care must be taken, since an excessive number of iterations can lead to overfitting, which will worsen the results.
- Gradient boosting: in this method, as in the previous, the next model in combination with the previous one produces better results than the independent models. The difference is that in this method not all the errors of the previous model are used to train a new model; the data that minimizes the error is sought. An example of this technique is XGBoost Chen and Guestrin (2016).
- Bagging: with this technique each model learns the error generated by the previous model using another subset for training Breiman (1996). The main difference with AdaBoost is the way in which the training subsets are selected. In AdaBoost, a sequential system is followed. With the errors of the previous model, a new model is obtained that solves them. This way, in Boosting (or AdaBoost) the actual model learns from the errors of the previous model. In Bagging, independent training subsets are selected. Finally, their predictions are combined using a voting technique. Since models trained with different subsets, each model have different errors. The errors of one model are compensated with the other models.

  One example of Bagging is Random Forest. Random Forest is a machine learning algorithm based on decision trees that combines several trees to produce a single strong model. It can be used as a classification algorithm Breiman (2001). It uses bagging to create each tree, with the objective of creating an uncorre-

lated forest of trees. The predictions of the forest are more accurate than the predictions of individual trees.

The techniques mentioned above can be applied in several machine learning tasks, but some specific to object detection. An object detector can be seen as a function that, given an image I, returns a list of detections, $D = [d_1, ..., d_N]$, in which each detection, d_i, is composed of $[b_i, c_i, s_i]$, b_i being the bounding boxes coordinates, c_i the object class and s_i the confidence score. In the ensemble of object detection models, given an image I, a list $DL = [D_1, ..., D_M]$ is returned with $i \in \{1, ..., M\}$, M being the number of models used. Once all the models produce their predictions (DL), several strategies can be carried out Casado-García and Heras (2020):

- Affirmative or OR ensembling: all detection lists (D_i) in the ensemble list (DL) are used. This means that if one of the models makes a prediction, it is considered correct.
- Consensus: if M models are used, to consider a prediction as correct it is necessary that $M/2 + 1$ models have made that prediction.
- Unanimous or AND ensembling: if M models are used, to consider a prediction as correct it is necessary that M models have made that prediction.

Once one of these strategies has been applied, it is common to make redundant predictions. This is because the object detector algorithm can make several predictions for the same object. Obviously, only the best one is interesting since keeping the rest would negatively affect the evaluation metrics and the results in production. For this reason, the next step is to apply NMS (Non-Maximal Suppression) to eliminate redundant predictions. NMS removes redundant predictions based on the confidence score Bodla et al. (2017). One alternative to NMS is Weighted-NMS Solovyev, Wang, and Gabruseva (2021). If there are redundant predictions, NMS selects the one with the highest confidence. Weighted-NMS takes all predictions into account, generating a new prediction from them. To produce the best possible predictions, the predictions with the highest confidence are given more weight. Weighted-NMS can produce better results than NMS if the quality of the predictions of each model is high.

All these techniques for the ensemble of deep learning models improve the results produced by the models independently of each other. But they have a drawback: they do not combine predictions from object detection and semantic segmentation. One of the requirements to apply Boosting, Bagging or these specific techniques for object detection is that they require data with similar characteristics to perform the ensemble process. However, the outputs provided by object detection and semantic segmentation are different. What would be feasible would be to apply Boosting or Bagging on different object detection and semantic segmentation models. Once the ensemble of both tasks has been performed independently, the ensemble of object detection and semantic segmentation could be performed with the technique proposed in this work. However, using Boosting or Bagging adds complexity to the models, and usually leads to overfitting. For these reasons, it was decided not to use them, and to apply the proposed technique directly.

In **?**, predictions are first made with YOLOv5. The output is then taken as input to UNET. However, it presents the problem that it depends on the architecture used. If in the future it is desired to use other segmentation or object detection algorithms, it will be necessary to modify the proposed architecture. The novelty of this work consists in creating a framework, as well as a methodology, that enables the integration of object detection and semantic segmentation independently of their architecture. This way,

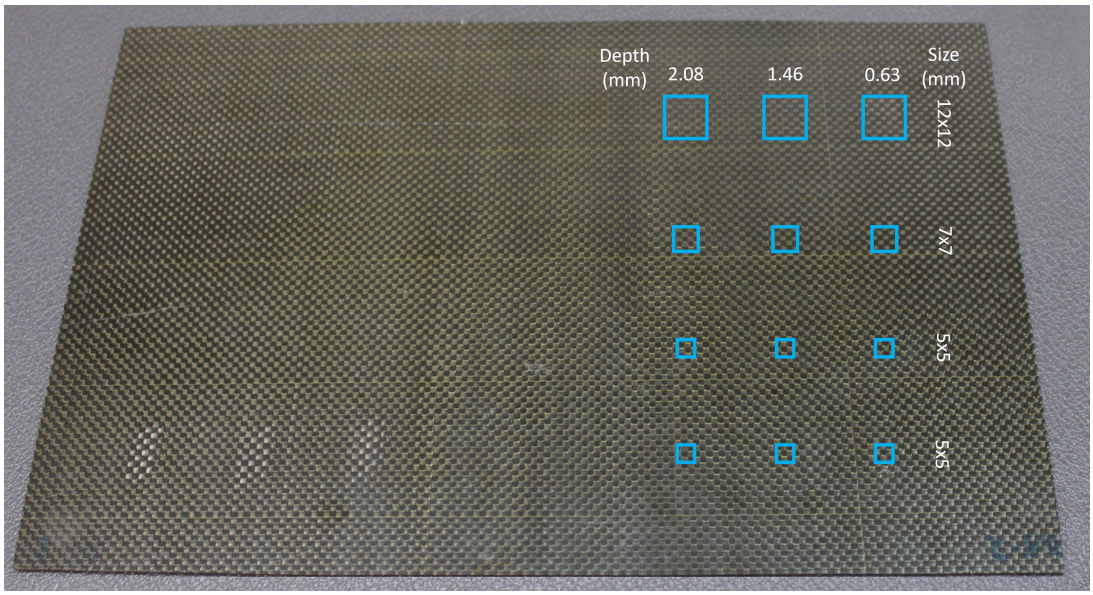

Figure 1. Carbon fiber solid laminate sheet

the reliability of predictions increases.

2. Materials and methods

2.1. *Dataset*

To obtain a dataset on which to apply the deep learning models, a carbon fiber solid laminate specimen is used. This laminate is composed of several layers of carbon fiber. Each layer contains fibers stacked and bonded by a resin. To simulate subsurface flaws, 12 defects of different sizes have been introduced. Figure 1 shows the specimen used.

To collect the images, several inspections are performed. In each inspection, the film is heated for 10 seconds, and then the heating is stopped for another 10 seconds. As shown in Figure 2, the infrared camera is setup at 1.5 m from the sheet and the lamps are located 1.6 m away.

Principal Component Thermography (PCT) is applied to reduce the noise generated during inspections. This technique simplifies the detection process, as it reduces the data variables. Each inspection has 1000 frames. After applying PCT, the 3 frames with the most relevant information are chosen to generate an RGB image, to feed the selected deep learning algorithms. This is because the object detection algorithms used take 3-channel images as input. Eq. 1 shows the signal-to-noise ratio (SNR) metric used to select the frames with the most relevant information. The SNR is calculated for each defect, where Def_u is the mean of the pixels in the defect area, Ref_u is the mean of the pixels in the non-defect area and Ref_σ is the standard deviation of the pixels in the non-defect area. Considering that in the images there is a part that does not belong to the sheet but to the laboratory, not all pixels can be used to calculate Ref_u and Ref_σ. For this reason, exclusion areas shown in Figure 3 have been used. Table 1 shows how the SNR is higher after applying PCT.

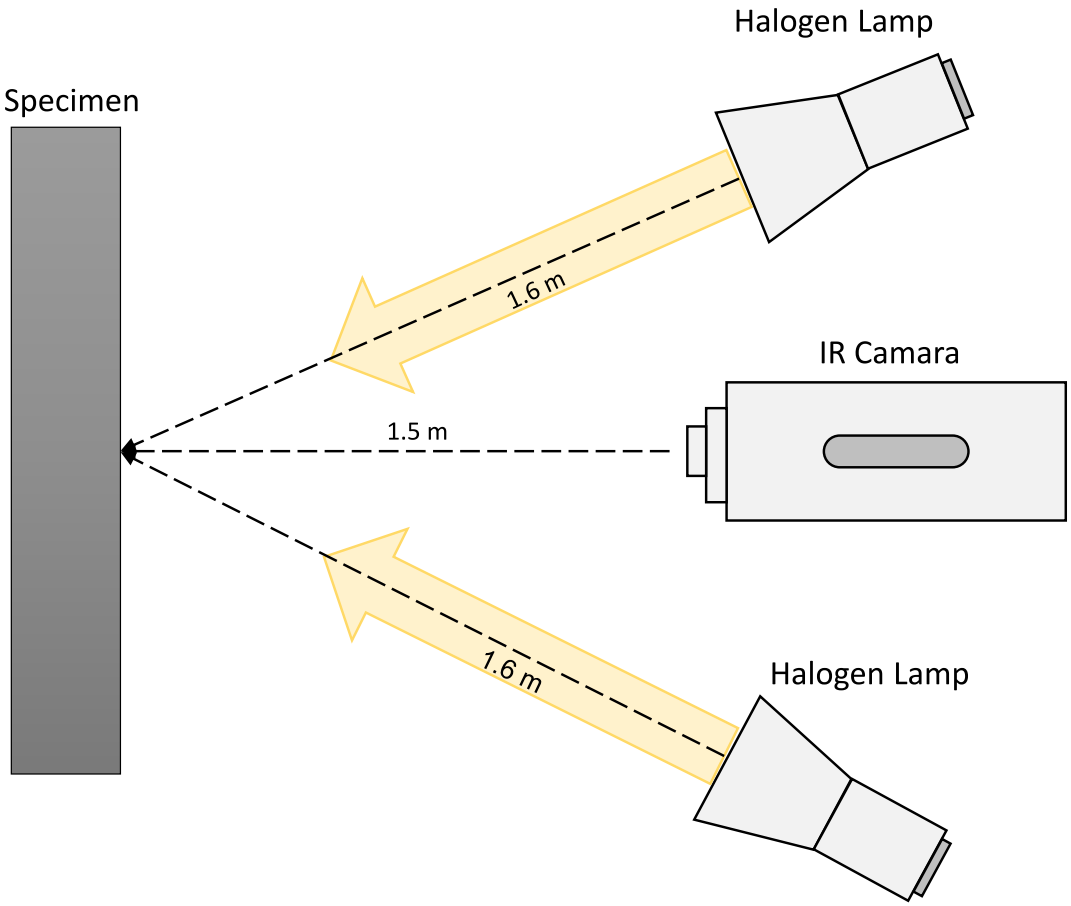

Figure 2. Infrared inspection setup

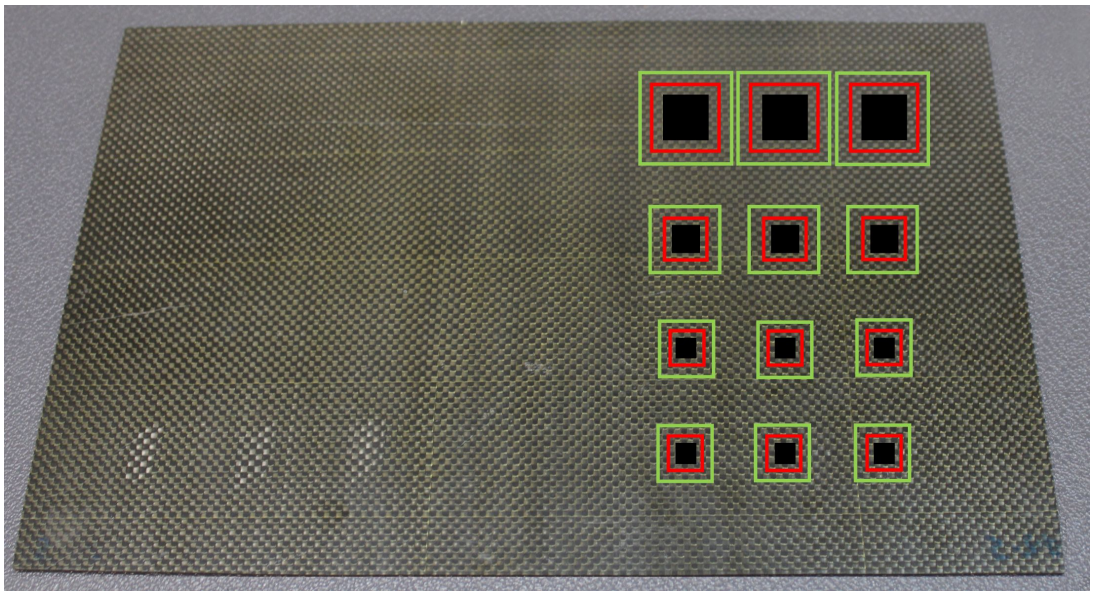

Figure 3. In black the defect area (Def_u), in red the exclusion area and in green the reference area (Ref_u and Ref_σ)

Table 1. SNR for each of the flaws before and after applying PCT.

	Before PCT	**After PCT**
Flaw 1	-3.6	-1.3
Flaw 2	-18.1	-5.7
Flaw 3	1.3	19.5
Flaw 4	-8.2	-1.6
Flaw 5	-14.9	-9.6
Flaw 6	-8.7	11.7
Flaw 7	-31.6	-7.8
Flaw 8	-14.5	-17.2
Flaw 9	-13.0	9.2
Flaw 10	-7.7	-22.5
Flaw 11	-24.8	-5.2
Flaw 12	-3.7	3.8

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

After all this process, 36 images of 640x480 size are obtained. Twenty-four are used for training and twelve for testing. Before using them to train deep learning based models, it is necessary to label them. Since the lab illumination can lead to confusion, the CLAHE algorithm (Contrast Limited Adaptive Histogram Equalization) Reza (2004) is used to improve the contrast. After this preprocessing step, images are labeled. This can be an arduous task, therefore an autolabeling tool has been used. To use it, it is necessary to label the first image carefully. The rest of the images are rotations and translations, therefore it is possible to calculate where the defects are from the labeled image. Firstly, it is necessary to calculate the edges of the sheet. For this purpose, the

Canny algorithm Rong et al. (2014) is used. Then, the corners of the sheet are calculated. For doing so, the Hough transformation is used to look for lines on a binary edge image Ji and Qi (2011). The corners are calculated as the intersection of lines. Once the corners are calculated, it is possible to determinate the correspondence between the first labeled image and the rotated. Thanks to this correspondence, it is possible to know where the defects are in the rotated images, and therefore it is not necessary to label them manually.

2.2. *Object detection techniques based on CNN*

In recent years, several object detectors have given good results in popular datasets such as COCO Lin et al. (2014). These object detector algorithms belong to one of these two types: one-state and two-state detectors. Two-state detectors, such as R-CNN Girshick et al. (2014), propose a set of regions of interest (ROI) and then classify each of these ROIs. One-state detectors, such as SSD Liu et al. (2016) or YOLO Redmon et al. (2016), process the whole image at once. Normally, one-stage detectors are faster, therefore, in this work the YOLO (You Only Look Once) family of detectors are used.

a) YOLOv1 Redmon et al. (2016): the first version of the YOLO family introduces inference in real time to the CNN-based object detectors. For this, YOLO divides the image into an $S \times S$ grid. Each cell of the grid is responsible for detecting the objects whose center falls in that cell. To reduce the number of redundant predictions, Non-Maximal Suppression (NMS) is applied.
b) YOLOv2 Redmon and Farhadi (2017): the second version of YOLO introduces anchor boxes. In YOLOv1 it was necessary to train from scratch. With these anchor boxes, the model has some initial guesses, thus it does not start from zero. In other works, such as Faster R-CNN Ren et al. (2017), the anchor boxes were selected by hand, but in YOLOv2 they are selected using k-means. This produces better anchor boxes, and therefore, better results.
c) YOLOv3 Redmon and Farhadi (2018): the previous versions mainly failed in detecting small objects. YOLOv3 introduces predictions on three scales, which gives better results for small objects.
d) YOLOv4 and YOLOv5 Bochkovskiy, Wang, and Liao (2020); Jocher et al. (2021): these versions of this object detector are the continuation of YOLOv3 by different authors. Both detectors introduce several improvements widely used in modern CNN-based object detectors. The networks are divided in three parts: backbone, neck and head. CSPDarknet-53 is used as backbone, PA-NET as neck and YOLOv3 as head. Fig. 4 shows the architecture of YOLOv5. They also introduce novel techniques related with data augmentation, such as mosaic augmentation.

2.3. *Semantic segmentation techniques based on CNN*

Several semantic segmentation networks have also been created. Two of the most widely used by the scientific community have been selected for this work: U-Net and DeepLab.

a) U-Net Ronneberger, Fischer, and Brox (2015) owes its name to the U-shape

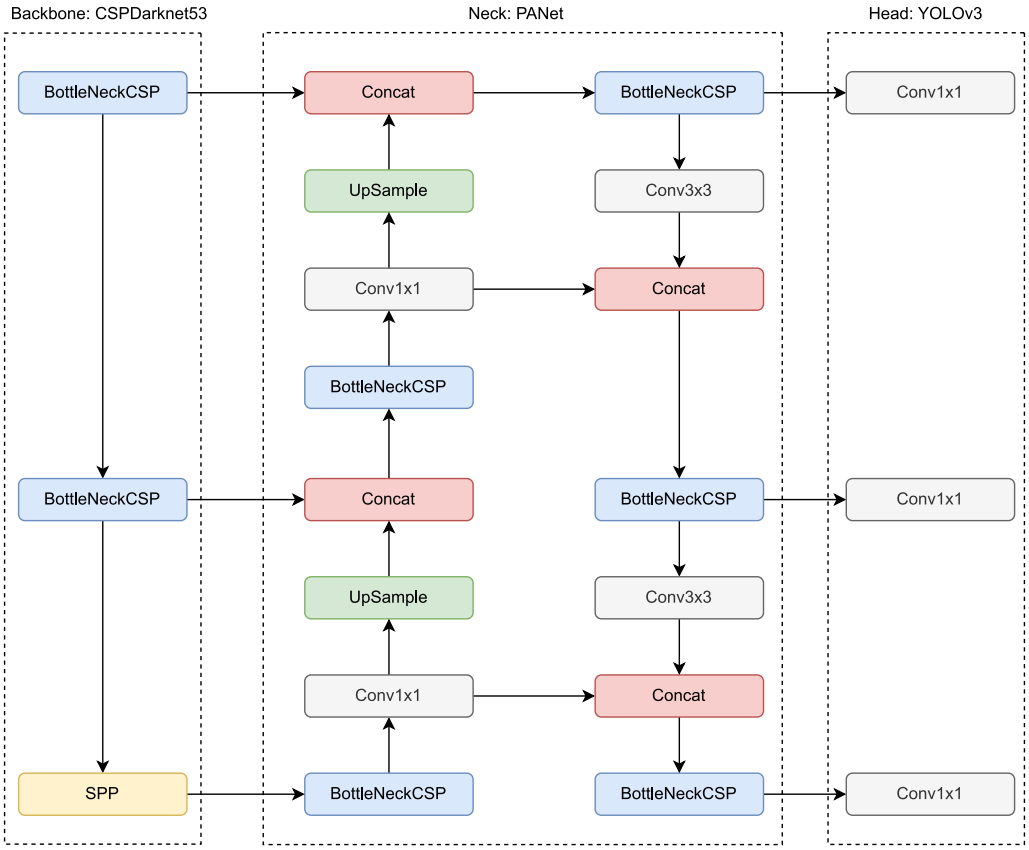

Figure 4. Architecture of YOLOv5. The model is composed of three parts: (1) the Backbone: CSPDarknet, (2) the Neck: PANet, and (3) the Head: YOLO Layer. The data goes through CSPDarknet for feature extraction initially, then it is processed by PANet for feature fusion. Finally, the YOLO Layer produces the detection results which includes class, score, location, and size.

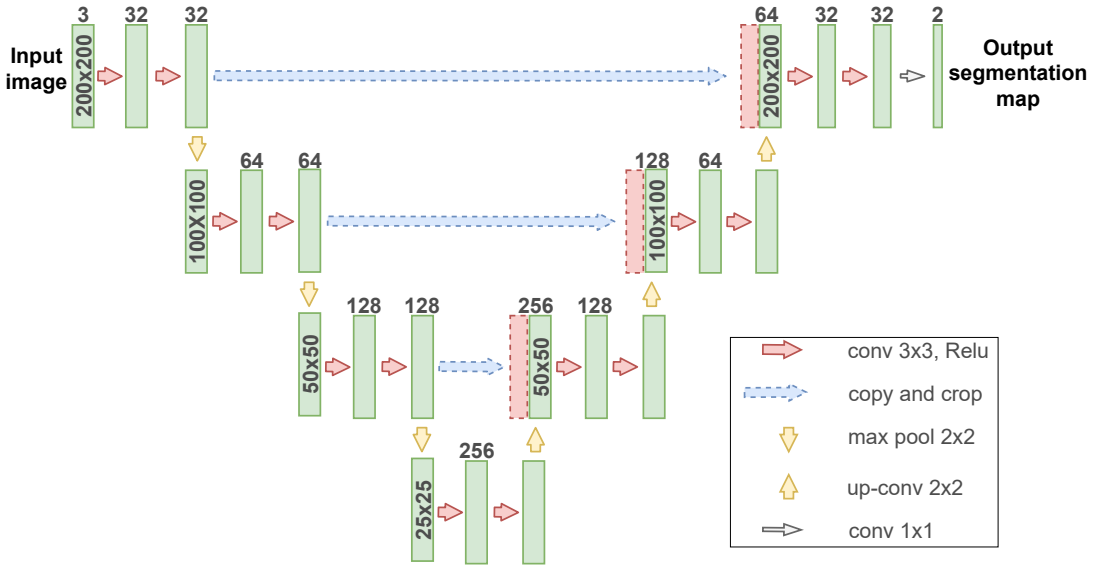

Figure 5. Architecture of U-Net. Each box represented in green corresponds to a feature map with multiple channels. The number of channels is indicated above the box. The size of the map in x-y dimensions is displayed within one of the boxes. The boxes represented in red indicate copied feature maps. The arrows indicate the various operations being performed.

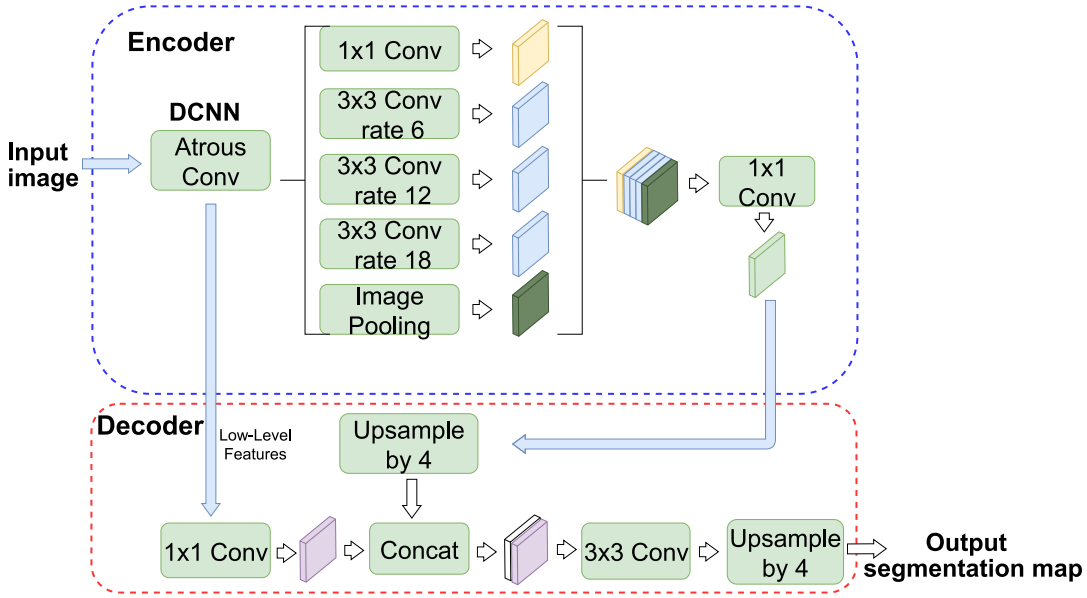

Figure 6. Architecture of DeepLabv3+. The encoder module captures multi-scale contextual information by using atrous convolution at multiple scales, while the decoder module, which is simple yet effective, improves the segmentation results along the boundaries of the objects.

encoder–decoder architecture, shown in Fig. 5. Originally, it was used for biomedical image segmentation. Its simplicity allows it to be adapted to different problems by modifying the number of classes or the size of the input images.

b) DeepLabv3+ Chen et al. (2018b) is a semantic segmentation network that arises as an evolution of previous networks. DeepLabv1 Chen et al. (2014) uses atrous convolutions to tune the resolution at which features are calculated. DeepLabv2 Chen et al. (2018a) uses ASPP (Atrous Spatial Pyramid Pooling) to improve the precision at different scales. DeepLabv3 Chen et al. (2017) updates the ASPP module and adds batch normalization. Finally, DeepLabv3+ adopts an encoder–decoder architecture, as shown in Fig. 6.

2.4. *Proposed techniques for semantic segmentation and object detection ensemble*

Improving the predicted mask from semantic segmentation is a complicated task. In this work, the improvement is proposed by combining semantic segmentation and object detection. Fig. 7 shows the proposed methodology. After collecting the data, it must be labeled properly for object detection and semantic segmentation. Then, after several training experiments in which the objective is to find the best hyperparameters, an object detection and semantic segmentation model is obtained. Hereafter, the training data is ensembled following the two proposed strategies. Each strategy is performed independently, and then the results are compared.

- Intersection: the intersection between the contours predicted by the semantic segmentation method and the bounding boxes of the object detector is the simplest of the two proposed strategies. Pixels outside the intersection are discarded, thus eliminating incorrect predictions. To carry out this ensemble strategy, it is necessary to set the minimum confidence threshold of the bounding boxes, since

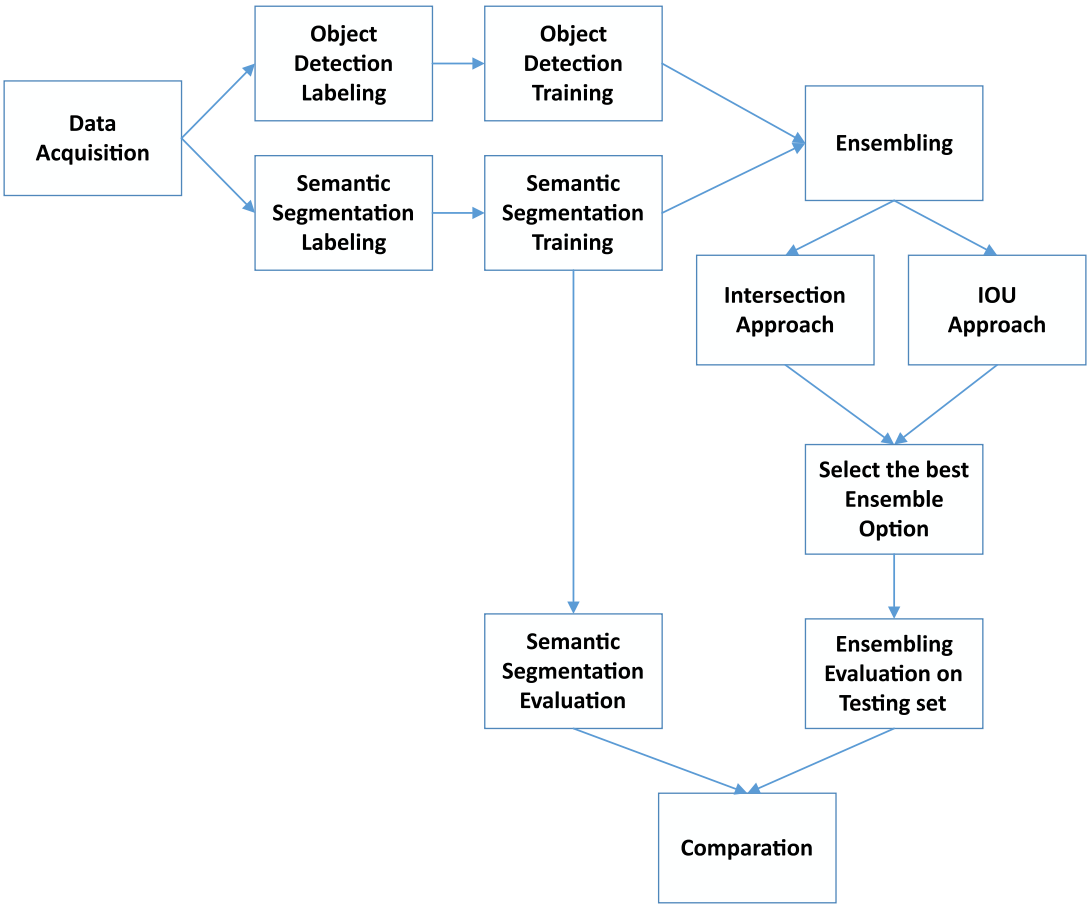

Figure 7. Proposed ensembling methodology.

the ensemble results will depend on it. To select the appropriate threshold, experimentation must be carried out using different thresholds to finally select the one that maximizes the evaluation metrics. In this work, the confidence thresholds belonging to the interval [0.1, 1] (with an step of 0.1) are used.

Fig. 8a shows the ground truth of an image, and Fig. 8b the predicted mask of that image. Visually, they are not 100% identical. To improve this, the intersecting pixels between the mask and the bounding box are calculated, as shown in Fig. 8c. As a result of this ensemble strategy, a new more precise mask is generated (Fig. 8d).

- IOU: intersection over the union or IOU, shown in Eq. (2), can be used to improve the quality of the predicted mask. This alternative is more complicated than the previous one, since it is not only necessary to evaluate different confidence thresholds, but also to evaluate different IOU thresholds. The higher the IOU between the contour and the bounding box, the higher the precision but the lower the recall. However, the lower the IOU, the lower the precision but the higher the recall. For this reason, it is necessary to find the IOU threshold that maximizes the evaluation metrics. This threshold, like the confidence threshold, depends on the quality of the predictions (both semantic segmentation and object detection). Therefore, it is necessary to evaluate it in each of the possible application scenarios.

  Fig. 8e shows the ensemble using this approach. The IOU between the contour in red and the predicted bounding box is higher than the IOU threshold established, but the IOU between the other contour and the bounding box is lower. Therefore, it is eliminated, as shown in Fig. 8f.

After the ensemble has been performed under the two strategies, the results are evaluated and the option that maximizes the evaluation metrics is selected. Then, the best ensemble option must be evaluated on the testing set and compared with the results of the semantic segmentation method. In section 3, it is shown how the application of these simple but effective strategies improves the precision obtained by semantic segmentation methods.

The advantages of this approach are the following: 1) the combination of the results of these two tasks makes it unnecessary to retrain a new model, 2) it is a simple solution that can be applied to any semantic segmentation and object detection method, 3) if future developments in the field of semantic segmentation or object detection are made, this method can be further implemented, and 4) unlike some instance segmentation methods, such as Mask-RCNN He et al. (2017), which have one branch for object detection and another for semantic segmentation, this method treats the two tasks independently. Therefore, if new ensembling strategies are proposed it will be easy to implement them.

2.5. *Evaluation metrics*

Although there are significant differences between the evaluation metrics in semantic segmentation and object detection, there are certain similarities. In both cases, the basic metrics are the following Hossin and Sulaiman (2015):

- True positives or TPs: number of correctly predicted elements.
- False positives or FPs: number of incorrectly predicted elements.
- False negatives or FNs: number of incorrectly unpredicted elements.

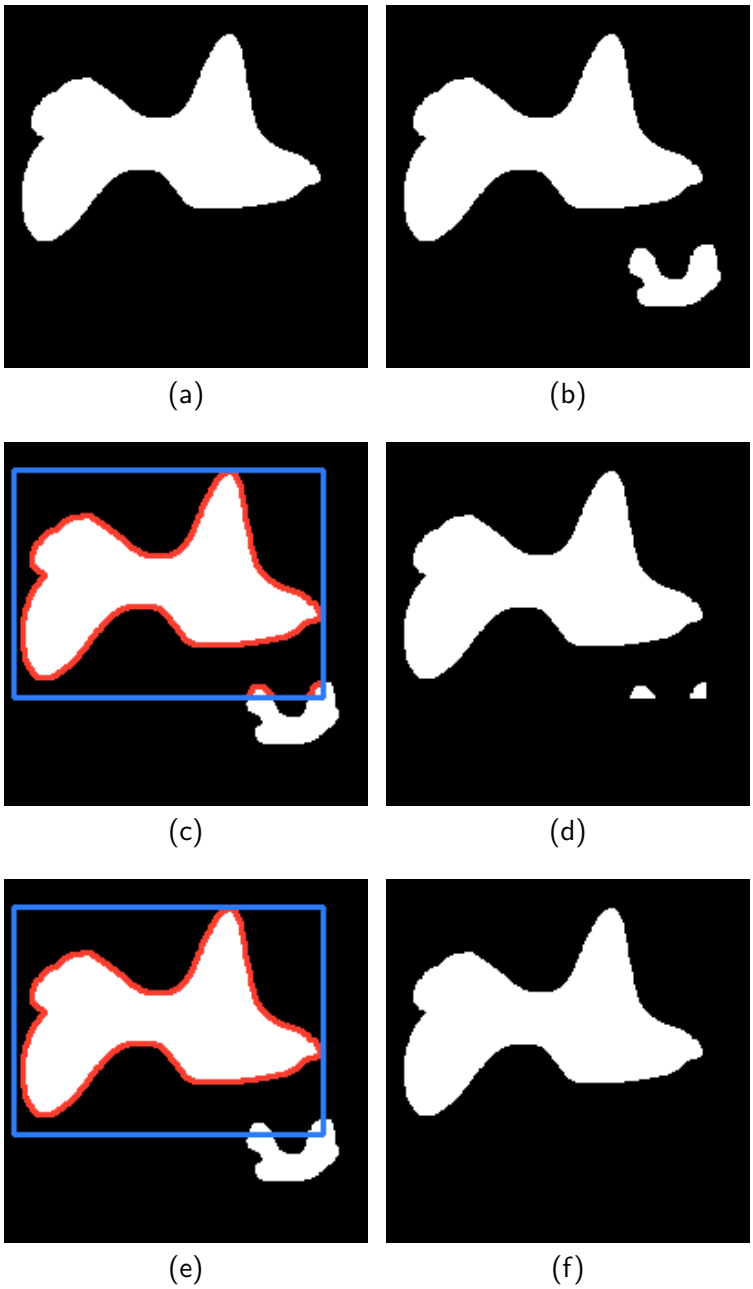

Figure 8. Proposed techniques for semantic segmentation and object detection ensemble. (a) Ground truth mask. (b) Predicted mask. (c) Ensemble using the intersection approach. (d) Resulting mask of using the intersection approach. (e) Ensemble using the IOU approach. (f) Resulting mask using the IOU approach.

- True negatives or TNs: number of correctly unpredicted elements.

In the case of semantic segmentation, the elements are the pixels of the image. For computing these metrics, it is only necessary to compare the ground truth mask with the predicted one.

In the case of object detection, the elements are the objects of interest. As it is common that a predicted bounding box does not match 100% with its corresponding ground truth bounding box, it is necessary to compare the degree of overlap between them. To compare them, the intersection over the union (IOU) is used, shown in Eq. (2). This is also known as the Jaccard index.

$$IOU = \frac{|P \cap G|}{|P \cup G|} \tag{2}$$

After calculating these metrics, it is possible to measure how good the performance of the model is. The precision, shown in Eq. (3), measures how good the predictions made are. The recall, shown in Eq. (4), measures how many of the elements are correctly detected. The F_1, shown in Eq. (5), weights the precision and recall in a single value.

$$Precision = \frac{TPs}{TPs + FPs} \tag{3}$$

$$Recall = \frac{TPs}{TPs + FNs} \tag{4}$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{5}$$

These are common metrics used in semantic segmentation and in object detection, but there are two other metrics commonly used in each field. The IOU, shown in Eq. (2) is used to calculate the area of overlap between two bounding boxes, but it can also be used to measure the degree of similarity between the ground truth mask and the predicted mask, in semantic segmentation Everingham et al. (2015). For this purpose, the IOU is calculated at pixel level as shown in Eq. (6). For object detection, the Average Precision (AP) is the most commonly used metric Everingham et al. (2010). In object detection, each prediction is composed of the bounding box, the object class, and a confidence value. This confidence value expresses how sure the model is about the prediction. For computing the metrics (TPs, FPs, FNs and TNs), it is necessary to establish a minimum confidence threshold. If the confidence value of a prediction is lower than the threshold established, it is discarded. This makes the precision and recall dependent on the confidence. Therefore, the AP is used. The AP is calculated as the area under the precision–recall curve Padilla et al. (2021). The curves are created using different confidence thresholds, which makes the AP non-dependent on the confidence. There are several ways of referring to the AP: 1) AP_{50} when a fixed IOU threshold of 0.5 is used to calculate the metrics, 2) AP_{75} when 0.75 is used as

Table 2. 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

the IOU threshold and 3) AP when the values in the interval [0.5, 0.95] (step of 0.05) are used as IOU thresholds and the different APs obtained are averaged.

$$IOU = \frac{TPs}{TPs + FPs + FNs} \tag{6}$$

3. Results and discussion

To perform the ensemble of object detection and semantic segmentation, it is first necessary to have the best models of both techniques independently. For this purpose, a large number of experiments are carried out, which are detailed below. All experiments have been performed 10 times. The results shown are the average. To check whether the results are statistically significant, the mean of the distributions is compared. In none of the tests performed at the 5% significance level is the null hypothesis rejected.

3.1. *Object detection experiments*

To detect subsurface defects, the following object detectors have been analyzed: SSD, YOLOv3, YOLOv4 and YOLOv5. After performing several experiments, it is concluded that 24 training images are not enough to achieve the expected results. For this reason, it was decided to apply a policy of data augmentation until obtaining 500 training images. Data augmentation consists of these modifications: 1% to 20% zoom variations, horizontal and vertical flips, $\pm 5°$rotation, and random elastic image distortion.

After performing several experiments with each of the detectors, it is concluded that YOLOv5 offers the best results. Table 2 shows the confidence threshold used with each of the algorithms. It is calculated from a validation subset. On that subset the metrics are calculated using all possible confidence thresholds in the interval [0, 1] with a step of 0.001. The confidence threshold that maximizes F_1 is the selected threshold. As IOU threshold, 0.5 has been selected for all algorithms. This is because experiments have been performed on the validation subset, and it has been concluded that this value does not affect the results. This is because there are no overlapping objects. Table 3 shows the hyperparameters used. The hyperparameters that maximize the results of each algorithm are shown. The number of epochs used in each algorithm varies because the training convergence times are not the same for all algorithms.

Figure 9 shows some examples of subsurface defect detection with YOLOv5. As can be seen in Figure 9c, all defects are detected. However, Figure 9d shows and example where two of the defects are not detected.

Table 3. 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	1 000	8	0.001	CSPDarknet53	adam

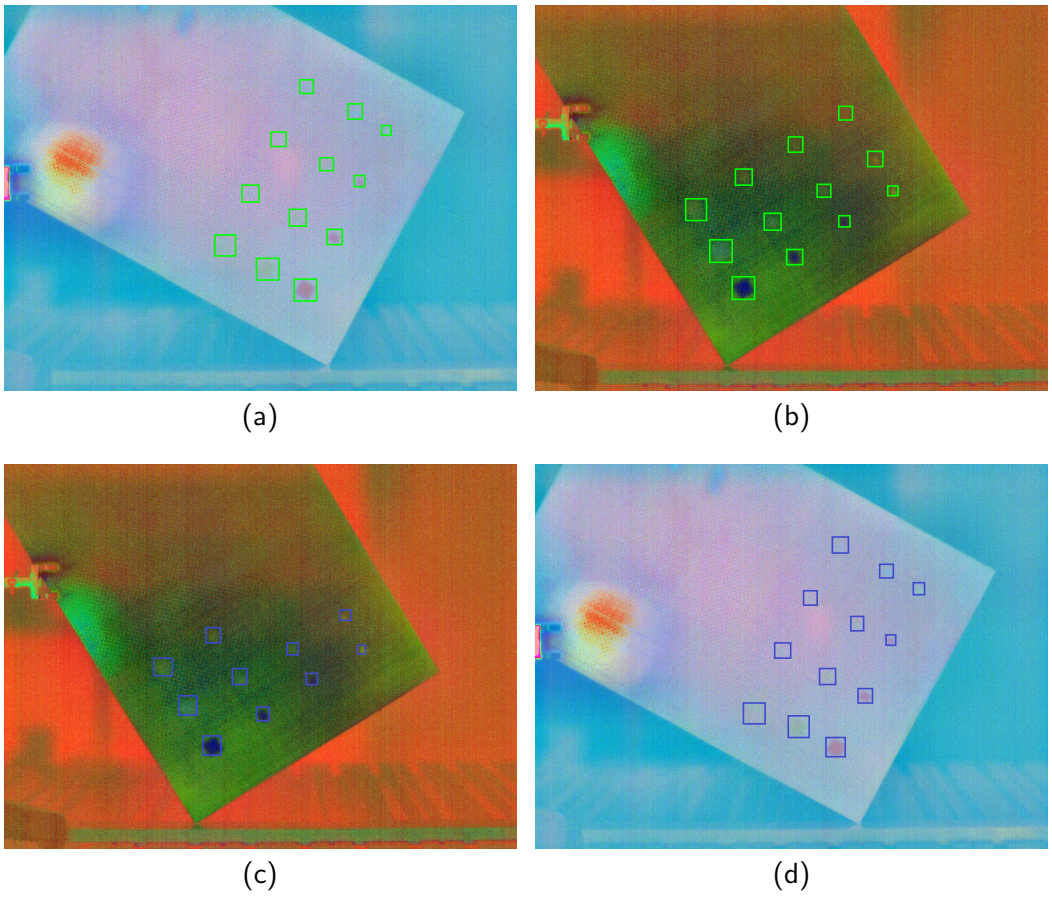

Figure 9. Detections performed with YOLOv5. (a and b) Ground truth. (c and d) Detections with YOLOv5.

Table 4. Results obtained with UNet and DeepLabv3+.

	Channels	**Precision**	**Recall**	F_1	**IOU**
UNet	3	0.689	0.717	0.703	0.542
UNet	30	0.764	0.726	0.745	0.593
DeepLabv3+	3	0.760	0.786	0.773	0.629

3.2. *Semantic segmentation experiments*

The following semantic segmentation algorithms have been used: UNet and DeepLabv3+. UNet is a network that allows the use of images with multiple channels, therefore more information can be used than with other networks. For this reason, several experiments are performed using images with 3 and 30 channels. Since the 30-channel images contain more information, the results are better. However, DeepLabv3+ gives better results than UNet, despite using 3-channel images. Table 4 shows the results obtained.

Figure 10 shows some examples of predictions made with these networks. UNet with 3 channels achieves good results, however using 30 channels generates less FP (False Positive). Even so, DeepLabv3+ achieves the best results.

3.3. *Model ensemble experiments*

The objective of this work is to increase the reliability of subsurface defect detection. In other words, to improve precision without negatively affecting recall. To this end, it is proposed the ensemble of object detection and semantic segmentation models using two approaches: intersection and IOU.

In the case of IOU, it is necessary to establish the minimum confidence threshold required in object detection, as well as the level of IOU between the predictions of object detection and semantic segmentation required to accept a prediction as correct. As the confidence level increases the precision increases. However, recall decreases negatively affecting F_1. For this reason, a series of experiments to find out the optimal value are carried out. Something similar happens with the IOU. As the IOU threshold increases so does the reliability of the predictions, however too many predictions that are actually correct are discarded. In the case of intersection, only the confidence threshold of object detections needs to be selected.

After performing a large number of experiments, where all possible combinations are tested, the conclusion is that for this scenario the best ensemble option is the intersection. With this approach, the best results are obtained with a confidence level between 0.1 and 0.7. This way, the original results shown in Table 5 are slightly improved. In Figure 11 the AUPRC (Area Under the Precision–Recall Curve) is shown. The training performed with DeepLabv3 with this dataset lasts 1 hour and 18 minutes. However, the training with YOLOv5 takes 8 minutes. YOLOv5's short trainings make ensemble processing inexpensive, regardless of the required time.

Figure 13 shows some examples where it can be seen how the predicted mask after ensemble is more accurate than the one obtained directly by DeepLabv3+.

3.4. *Experiments with other pieces*

To reinforce the results obtained in the previous section, two other pieces with subsurface defects are used. These pieces are composed of different materials than the

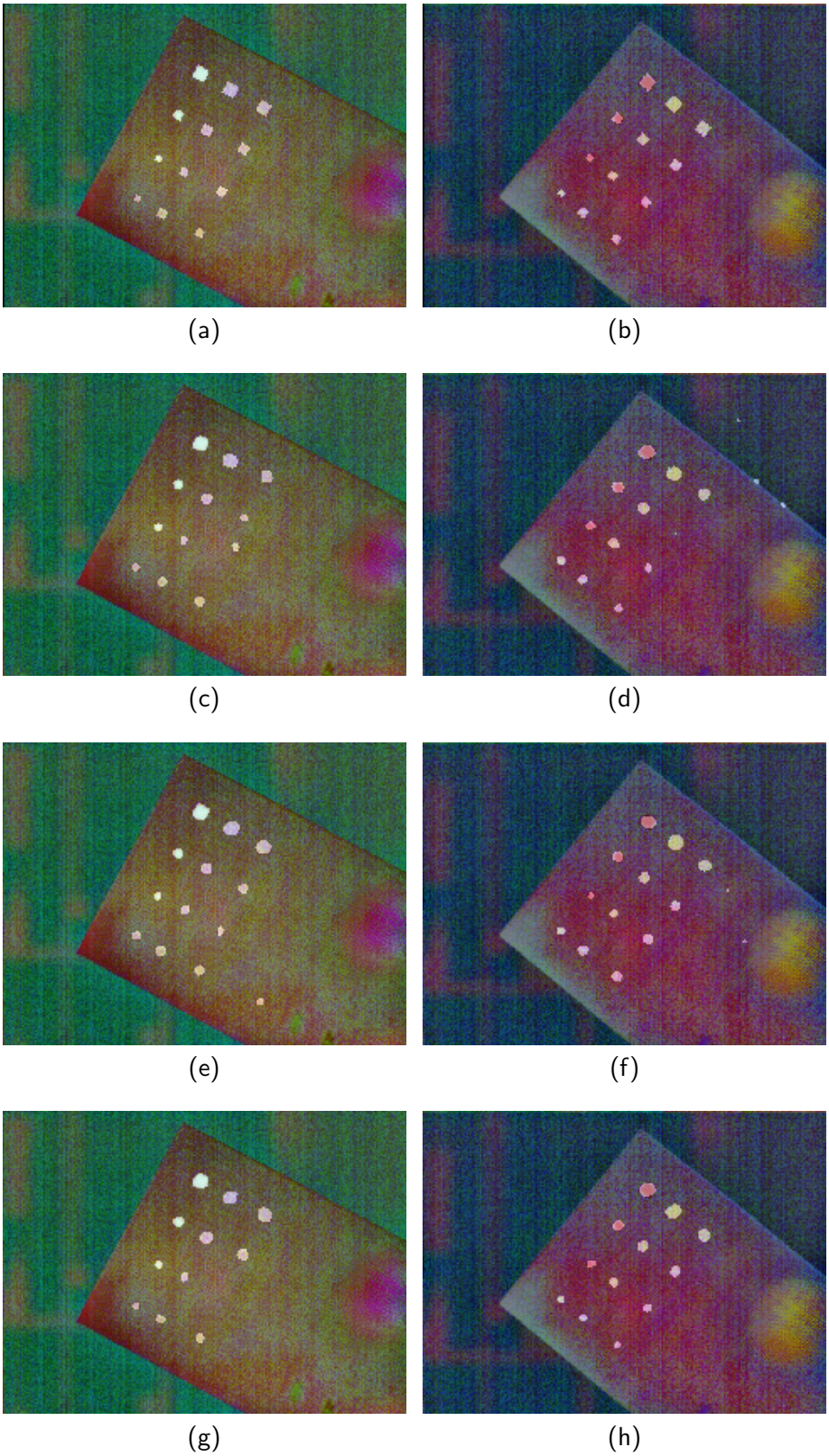

Figure 10. Examples of predictions made with semantic segmentation algorithms. (a, b) Ground truth. (c, d) Predictions with UNet 3 channels. (e, f) Predictions with UNet 30 channels. (g, h) Predictions with DeepLabv3+.

Table 5. Metrics after model ensembling with YOLOv5 and DeepLabv3+.

	Method	**Precision**	**Recall**	F_1	**IOU**
Original	-	0.7599	0.7858	0.7727	0.6295
Ensembled	IOU	0.7624	0.7858	0.7739	0.6312
Ensembled	Intersection	0.8102	0.7824	0.7960	0.6612

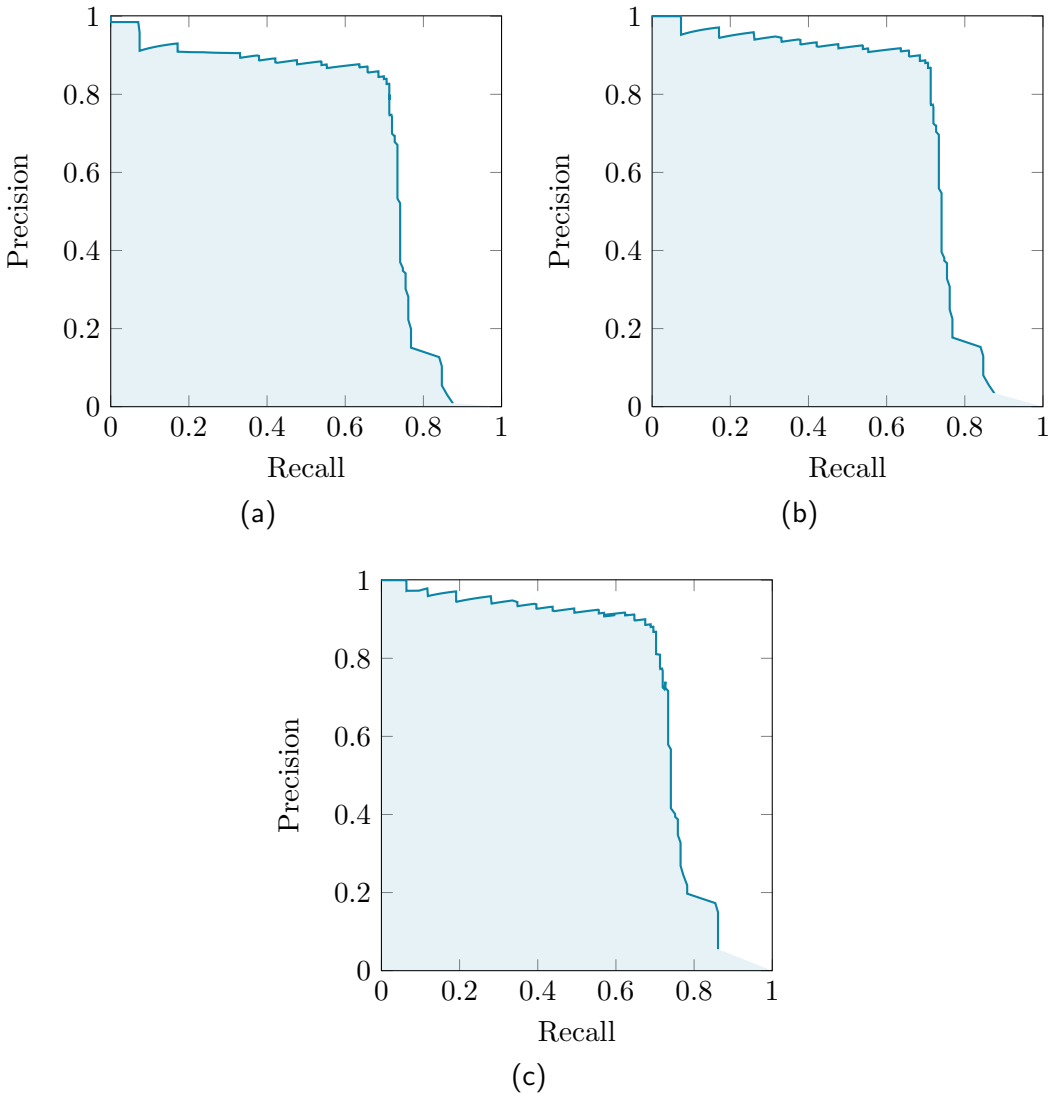

Figure 11. Area under the Precision–Recall curve (AUPRC) for DeepLabv3. (a) Before ensembling. (b) With IOU approach. (v) With intersection approach.

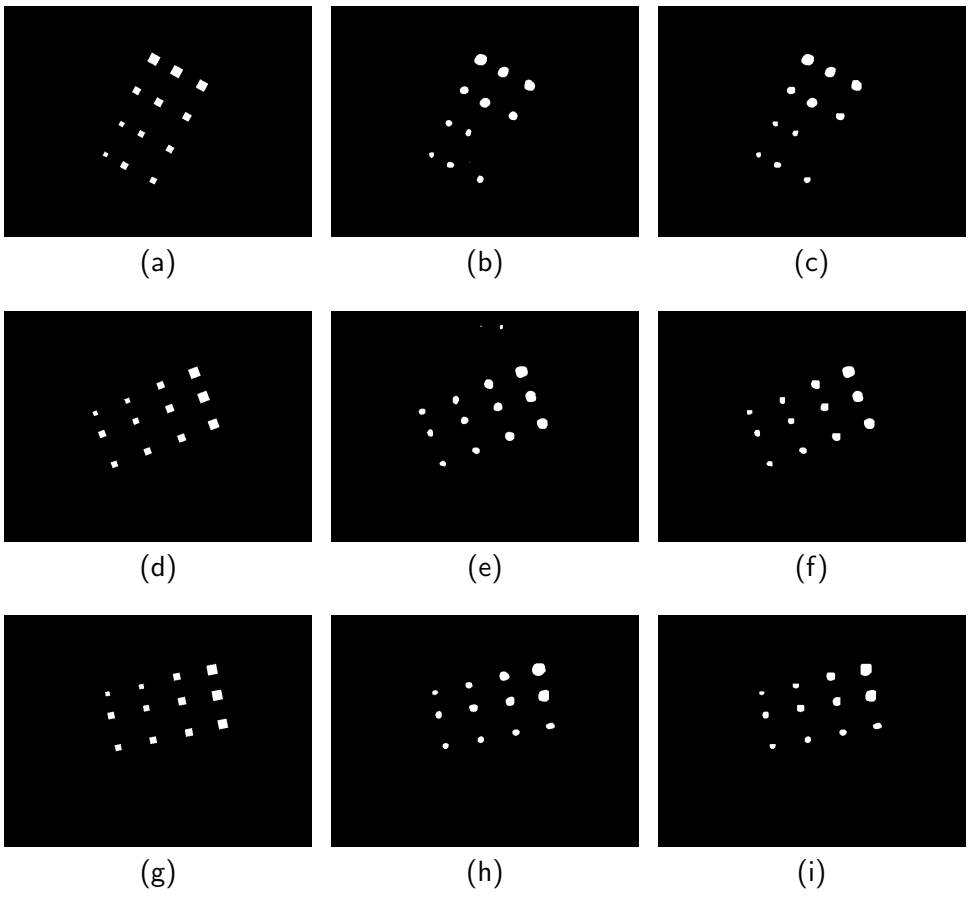

Figure 12. Examples of ensemble for subsurface defect detection. (a, d, g) Original masks. (b, e, h) Predicted masks with DeepLabv3+. (c, f, g) Masks after ensembling.

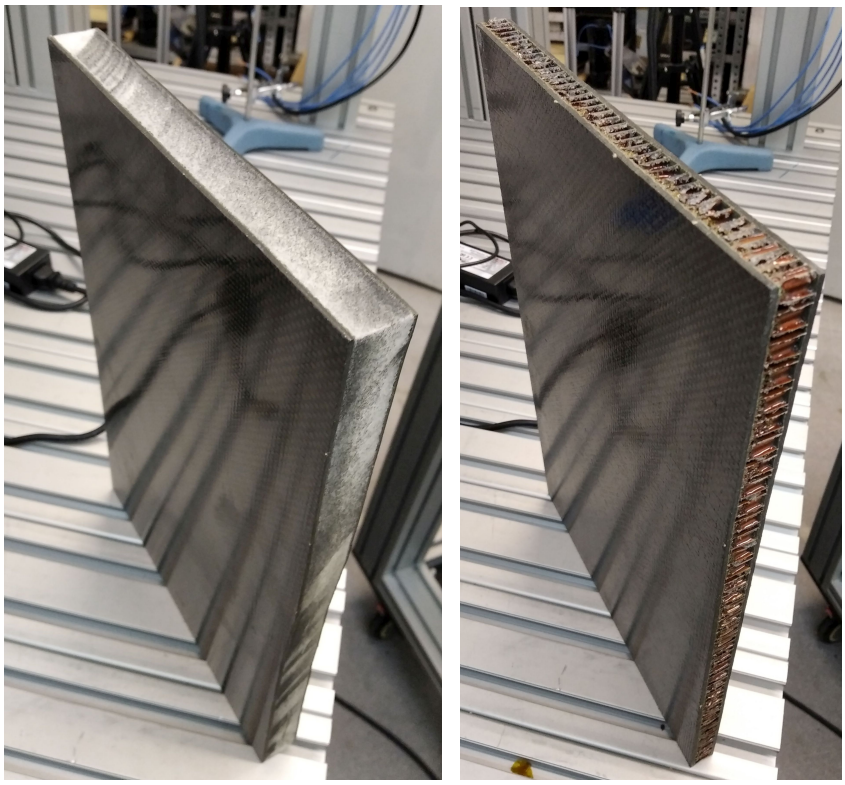

Figure 13. (a) Foam sandwich specimen. (b) Honeycomb sandwich specimen.

Table 6. Results obtained with UNet 3-channels and DeepLabv3+ on the foam sandwich panel.

	Precision	**Recall**	F_1	**IOU**
DeepLav3+	0.7895	0.7718	0.7805	0.6401
UNet 3-channels	0.5978	0.5131	0.5522	0.3814

one used for training. Even so, the previously generated models are used to check how feasible is the generalization of results from one piece to others.

The first piece, shown in Figure 13(a), is composed of a foam sandwich specimen with flaws at different depths than the original sheet. The second piece, shown in Figure 13(b), is composed of a honeycomb sandwich specimen, also with flaws at different depths than the original piece. It is decided to use both pieces because of their wide use in the aeronautical and naval industry. These pieces are composed of a sheet similar to the original and a filler. The original sheet was composed of 12 layers, while these are composed of 20 layers. These modifications cause the defects to be found at a depth of 0.38 mm, 1.46 mm and 2.08 mm.

The results of applying DeepLabv3+ and UNet with 3 channels on the foam sandwich panel are shown in Table 6. DeepLabv3+ improves the results obtained with the original piece. This is partly due to the fact that only one image (one piece) is being used to evaluate the model, while in the original experiments several images (although all of the same piece) are used. With UNet, the results are worse than the original ones. This is logical since a different piece has been used than the one used for training.

Table 7 shows the results with the honeycomb sandwich panel piece. The results

Table 7. Results obtained with UNet 3-channels and DeepLabv3+ on the honeycomb sandwich panel.

	Precision	**Recall**	F_1	**IOU**
DeepLav3+	0.8705	0.3499	0.4991	0.3326
UNet 3-channels	0.5984	0.5509	0.5736	0.4022

Table 8. Results obtained after ensembling on the foam sandwich panel.

	Ensemble strategy	**Precision**	**Recall**	F_1	**IOU**
DeepLav3+	Intersection	0.8556	0.7523	0.8007	0.6676
DeepLav3+	IOU	0.7904	0.7718	0.7809	0.6406
UNet	Intersection	0.8416	0.4951	0.6234	0.4529
UNet	IOU	0.6019	0.5056	0.5496	0.3789

are lower than those obtained with the original piece. This is due to the fact that the material properties are not the same as those of the original piece, and therefore the information varies.

After performing the segmentation, these models are ensembled with YOLOv5 using the intersection and IOU approach. Tables 8 and 9 show the results of the ensemble on both pieces. In both cases, the results obtained in segmentation are significantly improved.

Figure 14 (a,b) shows the images of the alternatives pieces used with their corresponding ground truth. Figure 14 (c,d) shows the predictions made with DeepLabv3+ and Figure 14 (e,f) shows the resulting mask after ensembling YOLOv5 and DeepLabv3 using the intersection approach, since it gives highest F_1 than with the IOU. If Figure 14 (c,d) and Figure 14 (e,f) are compared with Figure 14 (a,b), it is clear that the ensembling process improves the resulting mask. For this reason, the F_1 for the foam sandwich panel is improved by 2%, and the F_1 for the honeycomb sandwich panel by 1%.

Figure 14 (g,h) shows the predictions of UNet. This results, are improved by the ensembling process, as shown in Figure 14 (i,j). With the intersection approach, the F_1 is improved by a 7% for the foam sandwich panel and for the honeycomb sandwich panel by a 4%.

4. Conclusion

The detection of defects in industry is of vital importance as the safety of people may depend on it. In recent years, several advances have been made in the field of defect detection in industrial sheets. Most of these works use semantic segmentation or object detection algorithms independently to detect such defects. In this work, the possibility of combining both approaches to improve the precision of the predictions is studied. For this purpose, a real industrial scenario has been selected where, thanks to

Table 9. Results obtained after ensembling on the honeycomb sandwich panel.

	Ensemble strategy	**Precision**	**Recall**	F_1	**IOU**
DeepLav3+	Intersection	0.9304	0.3499	0.5085	0.3409
DeepLav3+	IOU	0.8844	0.3426	0.4939	0.3279
UNet	Intersection	0.7161	0.5429	0.6176	0.4468
UNet	IOU	0.6796	0.5509	0.6085	0.4373

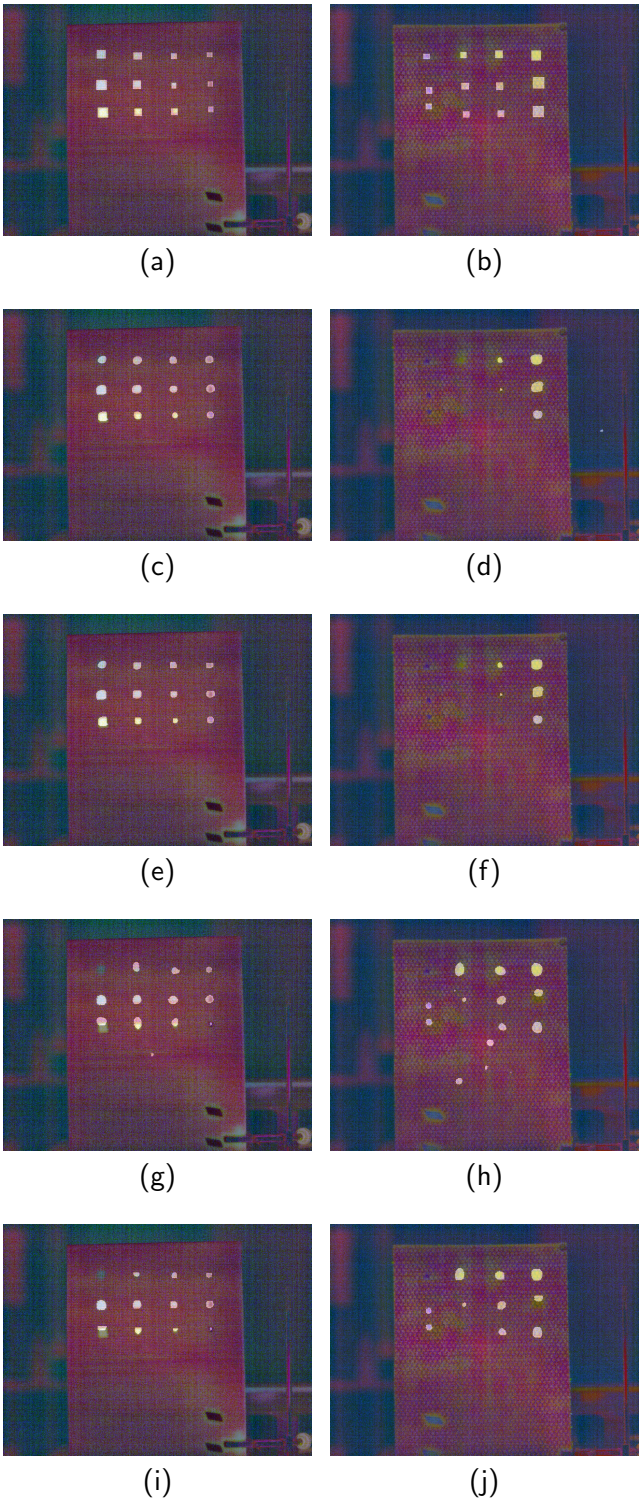

Figure 14. Detections performed after model ensembling. (a) Foam sandwich specimen ground truth. (b) Honeycomb sandwich specimen ground truth. (c) Predictions made with DeepLabv3+ on the foam sandwich panel. (d) Predictions made with DeepLabv3+ on the honeycomb sandwich panel. (e) Detections on the foam sandwich panel after ensembling YOLOv5 and DeepLabv3+. (f) Detections on the honeycomb sandwich panel after ensembling YOLOv5 and DeepLabv3+. (g) Predictions made with UNet on the foam sandwich panel. (h) Predictions made with UNet on the honeycomb sandwich panel. (i) Detections on the honeycomb sandwich panel after ensembling YOLOv5 and UNet. (j) Detections on the honeycomb sandwich panel after ensembling YOLOv5 and UNet.

the methodology proposed in this work, the precision is considerably improved while maintaining a similar recall. In the selected scenario, deep learning is combined with active thermography to identify subsurface defects in single carbon fiber solid laminate sheets.

The proposed methodology includes two ensemble strategies. The first one is based on the intersection of the contours predicted by the semantic segmentation method and the bounding boxes predicted by the object detector. In this case, it is necessary to select the confidence threshold that maximizes the evaluation metrics. The second strategy is based on calculating the intersection over union (IOU) between the contours and the bounding boxes. If the IOU is lower than a certain threshold, the contour is discarded, and both the confidence threshold and the IOU threshold that maximize the evaluation metrics must be selected.

Thanks to these simple but effective ensemble strategies, the precision obtained independently by the semantic segmentation methods is improved by a 5%, with negligible effect on the recall. This ensemble approach has the advantage that it is not necessary to retrain a model, but rather it takes advantage of two previously trained models. This offers great flexibility: if a new object detection or semantic segmentation algorithm that improves the current results emerges in the future, these ensemble techniques will still be valid.

The results of the ensemble are highly dependent on the data and the algorithms used. For this reason, two alternative pieces are also evaluated. The precision of the foam sandwich panel is improved by a 7% to 24% and for the honeycomb sandwich panel by a 6% to 11%.

By using these ensemble techniques, false positives (FP) generated by the semantic segmentation models are considerably reduced. In this way, models obtained improve precision, making the detection of subsurface defects safer and simpler. Thanks to this improvement in reliability, the increasingly stringent quality control requirements demanded by the industry can be met.

Acknowledgments

This research was funded by the project PID2021-124383OB-I00 of the Spanish National Plan for Research, Development and Innovation.

5. References

References

Bang, Hyun-Tae, Solmoi Park, and Haemin Jeon. 2020. "Defect identification in composite materials via thermography and deep learning techniques." *Composite Structures* 246: 112405. https://www.sciencedirect.com/science/article/pii/S026382232030146X.

Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. 2020. "Yolov4: Optimal speed and accuracy of object detection." *arXiv preprint arXiv:2004.10934* .

Bodla, Navaneeth, Bharat Singh, Rama Chellappa, and Larry S Davis. 2017. "Soft-NMS–improving object detection with one line of code." In *Proceedings of the IEEE international conference on computer vision*, 5561–5569.

Breiman, Leo. 1996. "Bagging predictors." *Machine Learning* 24 (2): 123–140. https://doi.org/10.1007/BF00058655.

Breiman, Leo. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32. https://doi.org/10.1023/A:1010933404324.

Casado-García, Ángela, and Jónathan Heras. 2020. "Ensemble methods for object detection." In *ECAI 2020*, 2688–2695. IOS Press.

Chen, Liang-Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 2014. "Semantic image segmentation with deep convolutional nets and fully connected crfs." *arXiv preprint arXiv:1412.7062* .

Chen, Liang-Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. 2018a. "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40 (4): 834–848.

Chen, Liang-Chieh, George Papandreou, Florian Schroff, and Hartwig Adam. 2017. "Rethinking atrous convolution for semantic image segmentation." *arXiv preprint arXiv:1706.05587* .

Chen, Liang-Chieh, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018b. "Encoder-decoder with atrous separable convolution for semantic image segmentation." In *Proceedings of the European conference on computer vision (ECCV)*, 801–818.

Chen, Tianqi, and Carlos Guestrin. 2016. "XGBoost: A Scalable Tree Boosting System." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, New York, NY, USA, 785–794. Association for Computing Machinery. https://doi.org/10.1145/2939672.2939785.

Cruz-Vega, Israel, Daniel Hernandez-Contreras, Hayde Peregrina-Barreto, Jose de Jesus Rangel-Magdaleno, and Juan Manuel Ramirez-Cortes. 2020. "Deep Learning Classification for Diabetic Foot Thermograms." *Sensors* 20 (6). https://www.mdpi.com/1424-8220/20/6/1762.

Estrada-Pérez, Leidy V., Sandra Pradana-López, Ana M. Pérez-Calabuig, María Luz Mena, John C. Cancilla, and José S. Torrecilla. 2021. "Thermal imaging of rice grains and flours to design convolutional systems to ensure quality and safety." *Food Control* 121: 107572. https://www.sciencedirect.com/science/article/pii/S0956713520304886.

Everingham, Mark, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. 2015. "The Pascal Visual Object Classes Challenge: A Retrospective." *International Journal of Computer Vision* 111 (1): 98–136. https://doi.org/10.1007/s11263-014-0733-5.

Everingham, Mark, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. 2010. "The Pascal Visual Object Classes (VOC) Challenge." *International Journal of Computer Vision* 88 (2): 303–338. https://doi.org/10.1007/s11263-009-0275-4.

Fang, Qiang, Clemente Ibarra-Castanedo, and Xavier Maldague. 2021. "Automatic Defects Segmentation and Identification by Deep Learning Algorithm with Pulsed Thermography: Synthetic and Experimental Data." *Big Data and Cognitive Computing* 5 (1). https://www.mdpi.com/2504-2289/5/1/9.

Freund, Yoav, and Robert E Schapire. 1997. "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting." *Journal of Computer and System Sciences* 55 (1): 119–139. https://www.sciencedirect.com/science/article/pii/S002200009791504X.

Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 580–587.

He, Kaiming, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. 2017. "Mask R-CNN." In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct.

Hossin, Mohammad, and Md Nasir Sulaiman. 2015. "A review on evaluation metrics for data classification evaluations." *International journal of data mining & knowledge management process* 5 (2): 1.

Inagaki, Terumi, Toshimitsu Ishii, and Toshikatsu Iwamoto. 1999. "On the NDT and E for the diagnosis of defects using infrared thermography." *NDT and E International* 32 (5): 247–257. https://www.sciencedirect.com/science/article/pii/S0963869598000590.

Izquierdo, Manuel, Miguel Lastra-Mejías, Ester González-Flores, John C. Cancilla, Miriam Pérez, and José S. Torrecilla. 2020. "Convolutional decoding of thermographic images to locate and quantify honey adulterations." *Talanta* 209: 120500. https://www.sciencedirect.com/science/article/pii/S0039914019311336.

Ji, Ronghua, and Lijun Qi. 2011. "Crop-row detection algorithm based on Random Hough Transformation." *Mathematical and Computer Modelling* 54 (3): 1016–1020. Mathematical and Computer Modeling in agriculture (CCTA 2010), https://www.sciencedirect.com/science/article/pii/S0895717710005212.

Jocher, Glenn, Alex Stoken, Ayush Chaurasia, Jirka Borovec, NanoCode012, TaoXie, Yonghye Kwon, et al. 2021. "ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support." October. https://doi.org/10.5281/zenodo.5563715.

Lema, Darío G., Oscar D. Pedrayes, Rubén Usamentiaga, Pablo Venegas, and Daniel F. García. 2022. "Automated Detection of Subsurface Defects Using Active Thermography and Deep Learning Object Detectors." *IEEE Transactions on Instrumentation and Measurement* 71: 1–13.

Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. "Microsoft COCO: Common Objects in Context." In *Computer Vision – ECCV 2014*, edited by David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, Cham, 740–755. Springer International Publishing.

Liu, Kaixin, Qing Yu, Yi Liu, Jianguo Yang, and Yuan Yao. 2022. "Convolutional Graph Thermography for Subsurface Defect Detection in Polymer Composites." *IEEE Transactions on Instrumentation and Measurement* 71: 1–11.

Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. 2016. "SSD: Single Shot MultiBox Detector." In *Computer Vision – ECCV 2016*, edited by Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, Cham, 21–37. Springer International Publishing.

Lyra, Simon, Leon Mayer, Liyang Ou, David Chen, Paddy Timms, Andrew Tay, Peter Y. Chan, Bergita Ganse, Steffen Leonhardt, and Christoph Hoog Antink. 2021. "A Deep Learning-Based Camera Approach for Vital Sign Monitoring Using Thermography Images for ICU Patients." *Sensors* 21 (4). https://www.mdpi.com/1424-8220/21/4/1495.

Padilla, Rafael, Wesley L. Passos, Thadeu L. B. Dias, Sergio L. Netto, and Eduardo A. B. da Silva. 2021. "A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit." *Electronics* 10 (3). https://www.mdpi.com/2079-9292/10/3/279.

Pedrayes, Oscar D., Darío G. Lema, Rubén Usamentiaga, Pablo Venegas, and Daniel F. García. 2022. "Semantic segmentation for non-destructive testing with step-heating thermography for composite laminates." *Measurement* 200: 111653. https://www.sciencedirect.com/science/article/pii/S0263224122008624.

Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. "You Only Look Once: Unified, Real-Time Object Detection." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 779–788.

Redmon, Joseph, and Ali Farhadi. 2017. "YOLO9000: Better, Faster, Stronger." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July, 7263–7271.

Redmon, Joseph, and Ali Farhadi. 2018. "Yolov3: An incremental improvement." *arXiv preprint arXiv:1804.02767* .

Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. 2017. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (6): 1137–1149.

Reza, Ali M. 2004. "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* 38 (1): 35–44. https://doi.org/10.1023/B:VLSI.0000028532.53893.82.

Rong, Weibin, Zhanjing Li, Wei Zhang, and Lining Sun. 2014. "An improved Canny edge detec-

tion algorithm." In *2014 IEEE International Conference on Mechatronics and Automation*, 577–582.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. 2015. "U-Net: Convolutional Networks for Biomedical Image Segmentation." In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, edited by Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, Cham, 234–241. Springer International Publishing.

Solovyev, Roman, Weimin Wang, and Tatiana Gabruseva. 2021. "Weighted boxes fusion: Ensembling boxes from different object detection models." *Image and Vision Computing* 107: 104117. https://www.sciencedirect.com/science/article/pii/S0262885621000226.

Usamentiaga, R., P. Venegas, J. Guerediaga, L. Vega, and I. López. 2013. "A quantitative comparison of stimulation and post-processing thermographic inspection methods applied to aeronautical carbon fibre reinforced polymer." *Quantitative InfraRed Thermography Journal* 10 (1): 55–73. https://doi.org/10.1080/17686733.2013.774623.

Usamentiaga, Rubén, Clemente Ibarra-Castanedo, Matthieu Klein, Xavier Maldague, Jeroen Peeters, and Alvaro Sanchez-Beato. 2017. "Nondestructive Evaluation of Carbon Fiber Bicycle Frames Using Infrared Thermography." *Sensors* 17 (11). https://www.mdpi.com/1424-8220/17/11/2679.

Yousefi, Bardia, Davood Kalhor, Rubén Usamentiaga Fernández, Lei Lei, Clemente Ibarra Castanedo, Xavier PV Maldague, et al. 2018. "Application of deep learning in infrared non-destructive testing." *QIRT 2018 Proceedings* .