RESEARCH PAPER

Automated defect detection in digital radiography of aerospace welds using deep learning

Topias Tyystjärvi[1,2] · Iikka Virkkunen[1] · Peter Fridolf[3] · Anders Rosell[3] · Zuheir Barsoum[4]

Abstract

Aerospace welds are non-destructively evaluated (NDE) during manufacturing to identify defective parts that may pose structural risks, often using digital radiography. The analysis of these digital radiographs is time consuming and costly. Attempts to automate the analysis using conventional computer vision methods or shallow machine learning have not, thus far, provided performance equivalent to human inspectors due to the high reliability requirements and low contrast to noise ratio of the defects. Modern approaches based on deep learning have made considerable progress towards reliable automated analysis. However, limited data sets render current machine learning solutions insufficient for industrial use. Moreover, industrial acceptance would require performance demonstration using standard metrics in non-destructive evaluation, such as probability of detection (POD), which are not commonly used in previous studies. In this study, data augmentation with virtual flaws was used to overcome data scarcity, and compared with conventional data augmentation. A semantic segmentation network was trained to find defects from computed radiography data of aerospace welds. Standard evaluation metrics in non-destructive testing were adopted for the comparison. Finally, the network was deployed as an inspector's aid in a realistic environment to predict flaws from production radiographs. The network achieved high detection reliability and defect sizing performance, and an acceptable false call rate. Virtual flaw augmentation was found to significantly improve performance, especially for limited data set sizes, and for underrepresented flaw types even at large data sets. The deployed prototype was found to be easy to use indicating readiness for industry adoption.

Keywords Deep learning · Non-destructive evaluation · Welding · Data augmentation · Probability of detection

1 Introduction

Radiography is used extensively in inspections of castings and welds in the aerospace, nuclear and automotive industries. The main task is to find discontinuities that cannot be seen via visual inspection, like gas pores and embedded cracks, or surface breaking defects invisible to the naked eye. Jonsson et al. [24] provide guidelines to how weld imperfections affect fatigue strength. Because discontinuities reduce structural properties and may lead to unpredictable failure, non-destructive evaluation (NDE) has high requirements for reliability. The requirements are

especially strict in safety-critical components, like those used in aerospace.

In NDE of welds, radiography data is most commonly analysed by experienced human inspectors. This manual process is time consuming, operator dependent and expensive. Components often have large amounts of acceptable flaws, and very few unacceptable ones. The rarity of critical flaws and the monotony of the inspection data risk errors related to human factors. Bertović [4] summarizes human factors research in NDE. Traditionally, the individual skill of the inspector and psychological aspects like tiredness or stress have been considered to be the main human factors that affect NDE quality. The effects of inspection procedures, human-machine interactions and group influence on inspection reliability are increasingly considered. The development of highly capable automatic tools that work well with human operators is a key step in improving the reliability of NDE [5].

The workload of manual analysis in inspections are a limiting factor in terms of robustness and capacity.

✉ Topias Tyystjärvi
  topias@trueflaw.com

Extended author information available on the last page of the article.

Defect acceptance criteria may be related to size, shape, location and proximity between defects [20]. Complex criteria with precise size definitions pose the risk of operator subjectivity affecting the outcome, leading to inconsistent results. Moreover, because of the costly analysis, resources for radiography are often only allocated to the detection and classification of critical defects that require repair or rejection of the part. Collecting statistics of acceptable defects, like the average amount of porosity in welds, may be unexplored regardless of the potential benefit of more fine-grained quality control. With more economical and accurate analysis, improvements in manufacturing could be made just by analysing existing inspection data, which further drives the interest for automation.

Several automation techniques have been developed in the attempt to alleviate the difficulties of manual analysis, either partially or fully. Automated analysis relying on conventional computer vision or shallow machine learning [34, 36, 45] have seen some use, especially in casting inspections. They have not been widely adopted in welding applications due to challenges like varying component geometry, low or inconsistent contrast and brightness, and flaw-like anomalies or geometry that produce excessive false calls. Figure 1 shows examples of defects and potential false indications in radiography data from the material used in the current work.

Recently, a surge of interest has centred machine learning and deep learning for image analysis in particular. Deep learning models are developed by learning from massive sets of data containing thousands or millions of examples. Resources of annotated training data are limited for many NDE applications [11, 13], and thus, the potential of deep learning may not be fully utilized. Recent deep learning-based segmentation methods, like the U-net proposed by Ronneberger et al. [42], make it possible to develop well performing models using smaller data sets. Data augmentation or simulation methods are often used to compensate for the lack of natural training examples with promising results [11, 13, 26, 33]. Even with improving methodology, industry adoption may be delayed due to differences between the standard evaluation metrics in NDE and deep learning which make it difficult to show that sufficient performance has been achieved.

1.1 Weld defects

Welding processes can cause many different types of defects, the classification of which is standardized in for example ISO 6520-1:2007 [19]. Six imperfection categories are listed: cracks, cavities, solid inclusions, lack of fusion and penetration, imperfect shape and dimension, and miscellaneous imperfections. Some welding processes only risk the occurrence of some of these categories. This study is concerned with a welding process where the defects of interest are cracks and cavities. Cracks are linear imperfections in welds commonly caused by solidification or residual stresses. In many applications, cracks are a rare but most critical in terms of structural

Fig. 1 Weld radiography data features. (a) Non-defective crack-like anomaly (white arrow). (b) Small pore in a weld (white arrow), and lead letters with holes resembling pores (black arrow). (c) A pair of pores with unacceptable combined size (merged white circles), and smaller, acceptable pores (black circles). (d) A non-defective area on the edge of an imaging plate, resembling a large cavity (white arrow). (e) A chain of small pores (white arrows)
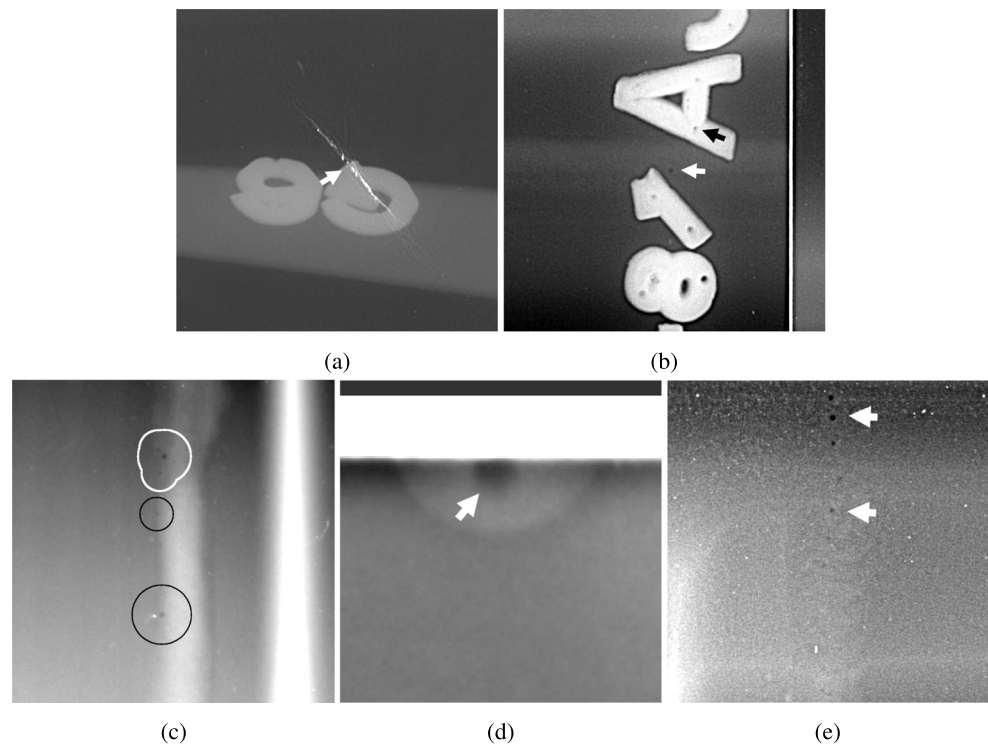
risk. Cavities are caused by trapped gas or shrinkage. Gas pores, characterized by round shape, are most common. ISO 5817:2014 [20] provides an example of acceptance levels for weld defects, often used in the industry as basis for more case-specific acceptance criteria.

1.2 Industrial radiography

Radiography is a common inspection method in manufacturing processes, especially castings and welds. An overview of the technique is provided in the ASM handbook by Greene et al. [14]. The three most prominent methods for capturing the image are film, computed radiography (CR) and digital detector array (DDA). Film radiography works similarly to regular film photography: Images are developed in a darkroom and viewed on an illuminating device with an adjustable back light or digitized. CR is based on storing the image a reusable imaging plate containing phosphor, which is then digitized by scanning with a laser [43]. DDAs directly produce a digital output from the radiography. Despite some differences in usage and image quality, film, CR and DDA techniques produce quite similar data, thus similar computer vision methods can be used for all of them. This study is conducted for a CR-based inspection.

ISO 10675-1:2017 [21] describes some limitations of radiography for defect detection. Surface imperfections like undercut or weld spatter may not be possible to evaluate due to geometry. Additionally, because radiography produces a 2D-image over the entire thickness of the studied part, thicker areas with similar volumetric imperfection density seem more severe. Cracks that open horizontally (parallel with respect to the imaging plate or film) are also undetectable due to the minute thickness of the crack not producing enough difference in intensity. These effects make the inspection more challenging. NASA-STD-5009B [37] provides values of minimum detectable crack sizes for radiographic NDE, depending on component thickness. The minimum detectable crack length for 2.72-mm-thick components is 3.8 mm, which is a suitable rough reference value for the application area in this work.

Greene et al. [14] summarize how the radiographic imaging process produces artefacts that reduce data quality. Shadows are formed due to the radiation point source: thick geometry casts a shadow away from the source because less of the beam is absorbed by the image conversion medium. Any geometry that is at an angle with respect to the imaging plate results in a distorted view. Many sources of unsharpness (blurring effects) exist, the most significant of which is usually geometric unsharpening, i.e. the partial shadow cast due to the width of the radiation source. Unsharpness effects limit the effective resolution, making small imperfections under some threshold undetectable. Radiation scatters when interacting with the test object or the surface directly under the object, contributing to blurriness and causing significant background noise. Small defects can be difficult to distinguish from this noise, exhibiting a small contrast to noise ratio (CNR).

To accurately capture small thickness differences, radiographic images are captured in high dynamic range. Twelve bit (4096 gray values) or 14 bit (16384 gray values) are most common, while images used in the large deep learning data sets like ImageNet [10] have 8 bit depth (256 values per channel). However, the theoretical dynamic range is often not utilized completely to avoid an excessively long exposure time. Areas outside the region of interest (ROI) like a bare imaging plate (maximum intensity) or lead lettering (minimum intensity) occupy the extreme ends of the dynamic range, while the interesting features are often small deviations in gray values. In components with varying thickness, like welds or non-planar geometry, flaw indications could be present on highly different ranges of intensities. This adds to the challenge of the operator, since absolute intensity cannot be used to determine the severity of the detected imperfection. Computer monitors can display 8 bits (256 gray values). High-end radiologist monitors can display 10 bits (1024 gray values) or even more, however, Kimpe and Tuytschaever [25] estimate that the human eye can only differentiate between about 900 gray values. To make use of the dynamic range that is larger than what the human eye or display can differentiate, operators repeatedly adjust the brightness and contrast levels to focus on small sections of the entire range at a time. The speed and quality of industrial radiography are in this sense limited by human physiology.

1.3 Automation in digital radiography NDE

Computer vision algorithms, and more recently deep learning, have been researched extensively to automate analysis in industrial radiography. The most common goals of the proposed methods are image enhancement, segmentation (marking the areas with defects), and classification between different flaw types. Compared to other fields of NDE like ultrasound or eddy current, radiography can more readily leverage developments in more general fields of automatic image analysis, since digital radiographs are essentially images with some differences as described in Section 1.2.

To compensate for the limitations in radiographic image quality, several enhancement and segmentation algorithms have been developed before the recent advances in deep learning. Edge detectors and sharpening filters are routinely used in industry. Nacereddine et al. [36] proposed image enhancement for digitized radiographs of welds by taking a user-defined ROI and applying median filtering for noise removal and contrast enhancement by look up table transformation. They studied different thresholding

approaches for defect segmentation, with image dilation and erosion post-processing. Schwartz [45] used a step wedge type penetrameter to link the luminosity of a digitized film X-ray image to material thickness. They segmented defects using Canny edge detection [7] and a threshold which was derived from the material thickness information, and reported high detection accuracy on X-ray images of welds and on a test object with drilled artificial flaws. This type of approach requires a uniform thickness and the use of specific equipment. In terms of industrial use, segmentation based on traditional computer vision is limited to cases where the acquired images do not display large variability.

Key developments in deep learning-based classification and segmentation by, e.g., Krizhevsky et al. [28] and Long et al. [31] have lead to machine learning reaching mainstream use in image analysis applications like self-driving cars [3]. Some differences exist between radiography and the common image analysis applications. In NDE, the detection of the objects of interest is the main focus and subsequent classification is more straightforward, because the objects are simpler and there are only a few categories present. In many cases, binary classification of background versus defect is sufficient, and further division can be achieved by observing the shape (aspect ratio) and size of defects via conventional computer vision. This is in contrast to the generic image classification, where the main challenge is to make sense of images with often a diverse range of features. The main challenges of deep learning-based radiography analysis are related to the small size and CNR of the defects, image quality limitations described in Section 1.2, and other features that are difficult to distinguish from the objects of interest (as shown in Fig. 1).

Deep learning approaches rely on a representative and large distribution of examples in order to achieve general knowledge on previously unseen input. As with many other fields, the use of machine learning for automated defect detection is limited by the available labeled training data. Especially some critical flaw types, like cracks, are relatively rare. Direct manual labeling becomes increasingly problematic with increasing data volume, because it is labour-intensive and susceptible to errors or inconsistency between labels. Mery et al. [35] published the GDXray database which contains about 20000 X-ray images in various categories. Notable for industrial radiography are 2727 images of castings and 88 images of welds. At present, it is the only publicly available database in this domain. This data set has been used for both early and more recent research in defect detection [32, 33, 44]. The castings set is suitable for deep learning bench marking, but the size of the welds data set is still limited in size.

Mery and Arteta [34] provide a good summary on the early automated classification of X-ray data using sparse representations, support vector machine (SVM) classifiers,

deep learning and other methods. They also extracted a voluminous data set of defected and non-defected image patches (32×32 pixels) from the GDXray database [35]. Working with small patches provides a way to increase the apparent data volume for the training set, but it also necessarily limits the network from making predictions based on larger scale features that may provide important context, like the area where the defect is located. They report high performance (95.2% accuracy) with hand-tuned extracted features and a SVM classifier. They also noted that the plain use of features extracted from convolutional neural networks (CNNs) pretrained in ImageNet (VGG, Simonyan and Zisserman [46]; AlexNet, Krizhevsky et al. [28]; and GoogleNet, Szegedy et al. [47]) did not work well for these X-ray images, despite being a successful tool for recognition in natural images. More recently, Du et al. [11] used a feature pyramid network (FPN) [30] approach to detect various defects in cast components. As in many cases, the data for training an X-ray detection system was limited and data augmentation (rotation, cropping and histogram equalization) was used to utilize the available data more efficiently. The authors also demonstrated that this conventional data augmentation has limited scope and additional data augmentation offers diminishing returns. Jiang et al. [23] proposed a novel pooling strategy for CNNs to better represent dark and light defects like slag and tungsten inclusion, and classified defects into six categories: crack, lack of fusion, lack of penetration, slag inclusion, porosity and non-defect. They used a set of 3486 32×32 pixel images divided into the six categories for training.

Simulated defects and physical artificial flaws have been used to generate training data with promising results. Recently, Gamdha et al. [13] used a combination of real and simulated synthetic X-ray images to train a mask R-CNN [17] segmentation network. Defects were introduced by simulating X-ray images from generated computer-aided design (CAD) models with shapes depicting characteristic flaws, like pores and voids. The addition of synthetic images improved the network performance and 87% accuracy was reported. Konnik et al. [26] proposed to address the same issue by making physical artificial flaws for computed tomography (CT). The artificial flaws were manufactured layer-by-layer by laser machining and then stacked for CT scanning, thus obtaining an accurately labeled 3D representation of defects. Mery [33] trained several modern deep learning classifiers on the GDXray [35] castings data set using simulated ellipsoidal defects superimposed onto defect-free areas, achieving high accuracy on true data.

1.4 Automatic radiograph evaluation in other fields

In the medical field, automatic X-ray analysis using deep learning has sparked broad interest. Rajpurkar et al. [38]

detected pneumonia from chest X-ray images using a DenseNet proposed by Huang et al. [18], along with class activation mapping developed by Zhou et al. [54] to achieve a coarse localization. They reported prediction scores surpassing the average score of experienced radiologists on a test set of about 400 images. Similarly, Li et al. [29] Used a combination of Resnet by He et al. [16], YOLO by Redmon et al. [40] and a fully convolutional network (FCN) by Long et al. [31] to annotate diseases in the Chestx-ray8 [51] data set. Successes in medical radiography suggest that good performance can also be achieved in industrial applications.

The medical X-ray applications centre around topics for which there are public data sets available of hundreds of thousands of images, like CheXpert [22] with 224,316 chest radiographs. Consequently, simulation or augmentation methods are not widely used. In NDE, the available data sets are usually much smaller. The largest public X-ray NDE data set, GDXray [35], only contains 88 images of welds. However, the detection task may also be simpler in nature, as defects can be more clearly characterized by shape and mainly vary in size and aspect ratio. Because NDE data are scarce but relatively simple, simulation methods and manufacturing artificial flaws [13, 26, 32] have been focused on in research.

1.5 Deep learning architectures for segmentation

Image analysis tasks can be divided into classification, where a label is assigned to an entire image without localization, and localizing tasks for which we use the same terms as He et al. [17]: object detection, where objects are marked by bounding box; semantic segmentation, where each pixel is classified to produce a fine-grained separation between classes; and instance segmentation, where (possibly overlapping) objects are individually detected in addition to the pixel-wise classification.

Commonly used architectures for classification include Resnet by He et al. [16] and Densenet by Huang et al. [18], that are characterized by very deep stacks of convolutional layers and an output vector describing predicted class probabilities. Zhao et al. [53] summarize recent developments in object detection models: For instance, YOLO networks by Redmon and Farhadi [39] and Bochkovskiy et al. [6] are commonly used. Long et al. [31] introduced FCNs for semantic segmentation, and greatly improved results in several tasks including PASCAL VOC [12]. The network consists of convolutional encoder and decoder stages with skip connections between them. Ronneberger et al. [42] developed the U-net for segmentation tasks on biomedical images. It is based on the FCN with the addition of more learnable upsampling layers and a weighted loss function for class balancing and instance separation. He et al. [17] proposed the Mask

R-CNN (based on Faster R-CNN by Ren et al. [41]), which outputs instance segmentation from region proposals generated by a CNN.

Jiang et al. [23] used a CNN with classification output for weld defect detection. Segmentation offers advantages over classification in NDE applications: Defects can be measured and counted, and their shape can be determined, which is important for defect acceptance as discussed in Section 1.1. Furthermore, segmentation models provide a more explainable prediction by default. With classification models, there is a risk of making seemingly correct predictions based on irrelevant features, like a scratch next to a crack. With a segmentation model, this would be exposed as only the pixels belonging to the scratch would be indicated. Object detection and instance segmentation architectures have also been used in NDE. Mery [33] used YOLO networks [6, 39] to output bounding boxes and Gamdha et al. [13] used mask R-CNN [17] for instance segmentation.

1.6 Data augmentation for segmentation tasks

Ronneberger et al. [42] developed the U-net for segmentation on biomedical images: cells in differential interference contrast and phase contrast microscopy, and neuronal structures in electron microscopic recordings. The tasks they showcase are similar to those in radiography NDE in that the data is 2-dimensional (can be described by a grayscale image) and the features of interest are less varied than in general image segmentation tasks. Moreover, the features differentiate from background faintly or have unclear, soft boundaries. When using small data sets, they found data augmentation by random elastic deformations significantly improved segmentation accuracy. The augmentations were made by generating deformation fields via interpolation from random displacement vectors and applying them on the training images.

Weld radiographs differ from the cell microscopy images by having a more varied background with background objects like geometry, markings, scratches and imaging artefacts that resemble the features of interest (defects). Moreover, straight lines like cracks or lack of fusion bear significance making random displacements less representative.

1.6.1 Virtual flaw data augmentation

Usually, augmentation consists of geometrical operations like flips, rotations, shear, random crops and scaling, and other image manipulations like random noise and brightness. In the context of NDE, conventional augmentation provides no variation for defect location. Simulation techniques, proposed by Gamdha et al. [13] and Mery [33],

achieve variation of shapes and locations. Simulation, however, relies on idealizations of true defects, potentially leading to the loss of some naturally occurring variability. Furthermore, imperfections with more complicated shapes, like cracks, lack of penetration in welds, or cracks stemming from pores, require an increasing effort to capture accurately.

Virkkunen et al. [49] introduced a method of creating virtual flaws for UT data to use in NDE qualification as alternatives to defects in physical mock-ups. The principle of the virtual flaw is to extract real flaw signals from inspection data, augment the flaw signal separately, and re-introduce it into another location. They generated a large set of data from limited real defects to more accurately measure inspector performance. Virkkunen et al. [50] used the virtual flaw technology as data augmentation for a deep learning classifier for ultrasonic testing (UT) data. Koskinen et al. [27] compared different types of virtual flaws and simulated flaws as training data for UT, finding that virtual flaws outperformed purely simulated data, and that the simulated data on its own was insufficient for generalization. The advantage of virtual flaw is that is generates variability to both background and flaw signals, while having both the defects and background drawn from real data. Xu et al. [52] used a somewhat similar technique to train a network for image matting, by extracting foreground objects manually from images with simple backgrounds and compositing them onto new backgrounds, reporting good performance on natural images.

Another key advantage of the virtual flaw is the ability to combine defects from several sources. For example, cracks are important to find yet very rare in actual data — by using virtual flaws, cracks can be extracted from other components or artificially manufactured cracks in validation samples, and transferred onto the case-specific background.

1.7 Validation

Evaluating the quality of inspection is an integral part of NDE. Overestimating capabilities is an obvious safety hazard, and underestimating leads to increased costs. The common measurements are reliability, accuracy, and false call rate. Reliability, the most important metric, is the ability to detect defects, and accuracy is the quality of determining the defect sizes. False call rate measures the frequency of making false indications.

A hit/miss probability of detection (POD) curve is a standard reliability measure in NDE. A comprehensive description is available in MIL-HDBK-1823A [1]. In a hit/miss POD evaluation, inspection data with known defect locations and sizes is inspected, documenting detected and undetected defects. A generalized linear model is then fitted to hit/miss data denoting the probability of detection with respect to defect size. A standard performance value obtained from the POD curve is the $a_{90/95}$, meaning the defect size that has 90% probability of detection with 95% confidence bounds. This measure is used to determine the smallest defect that can reliably be found. Notably, hit/miss POD does not take into account false calls, and assumes a steadily more difficult detection with smaller imperfection size. Other approaches for POD determination include â vs. a, described in ASTM E3023-21 [2] and model assisted POD [8, 9].

The false call rate is analysed separately from reliability. In MIL-HDBK-1823A [1], false call rate is defined as false positives divided by number of opportunities. The definition of opportunities is case-specific, and should be made in a way that reflects the inspection well.

Deep learning segmentation performance is often measured by intersection over union (IoU), which is the overlap of the true and predicted areas divided by the unions of the true and predicted areas. In contrast to the NDE metrics, IoU combines detection and false calls into a single value, and does not link object size to detection rate like POD. The IoU can also be calculated separately between each ground truth defect and overlapping prediction. A threshold is often set to determine the minimum amount of overlap required accept a correct segmentation. As discussed by Mery [33], the defects in radiographs can be very small, spanning only a handful of pixels, which makes the IoU threshold harder to reach than for most applications where the features of interest are larger. Another case is flaw clusters, where one prediction can correctly span two very close defects, but produce a low IoU score due to not giving two separate indications. Hit/miss POD does not necessitate correct size measurement, only detection.

1.8 Objectives of presented work

We identify the key issues in developing automated, deep learning-based systems for industrial radiography to be the scarcity of annotated data, specific challenges related to radiography data that differ from common images, and a need to adopt deep learning metrics to follow industry standards in NDE validation.

The contributions of this paper are as follows. First, we show that modern semantic segmentation networks can be used to find weld defects on a real inspection case, using CR data of welds in aerospace components. Secondly, we explore the added benefit of virtual flaw augmentation at different data set sizes, by first collecting and annotating a large data set and progressively testing performance on smaller subsets of the original material. Thirdly, we take metrics from standard practices in NDE and adapt them for a deep learning setting to facilitate deployment in industrial applications. Finally, we assess the effectiveness

of the system as an assisting tool in radiographic inspection by conducting a field experiment. We take the imaging process as given, i.e. changes to the data acquisition are not made. Performance is compared to evaluations by human inspectors, and not to, say, metallographic evaluation results.

2 Materials and methods

The experiment was divided into the following phases. First, raw data were gathered from a real radiographic inspection of aerospace welds and annotated manually. Additionally, sample plates with thermal fatigue cracks were imaged with the same radiography unit and annotated. Secondly, conventional and virtual flaw augmentation were utilized to generate training data sets from progressively smaller subsets of the entire data. A third, combined data set was created by equally sampling the two augmented data sets. Next, a modified U-net model [42] was trained using the different data sets to generate semantic segmentation masks of defects vs. background. Model performances were compared by using the relevant NDE metrics of POD, sizing accuracy and false call rate. After validation, the deep learning model for defect detection was deployed on a standalone device with a graphics processing unit (GPU). We created visualizations from the generated segmentation masks by applying size, shape and proximity criteria to separate acceptable and unacceptable defects. The system was tested by expert inspectors to qualitatively assess its readiness for industrial use.

2.1 Inspection case

The application investigated in this work is inspection of welds in aerospace components based on digital X-ray images. Flaws in the inspected welds are mainly small pores. Larger pores and pore clusters are the most common unacceptable imperfections encountered. The X-ray inspection is setup with CR and deploy source energies of roughly 150 kV and above. The CNR of the flaws can be very small, sometimes close to unity. Images have a high dynamic range which makes the inspection process arduous, as operators may need to navigate the images by zooming and manually adjusting brightness and contrast for different areas.

2.2 Raw data, annotations and manufactured cracks

A data set consisting of 223 CR images was collected, each containing several welded areas. The defects in the data set were mostly pores. To expand the training set with crack data, 5 thermal fatigue cracks were manufactured

onto samples of a similar material. To make the added data representative, the cracked samples were scanned with the same radiography equipment.

The annotation process is outlined in Fig. 2. The flaws were annotated manually for each image, resulting in 3500 separate indications, out of which 4 were cracks found in the original images and 5 were manufactured thermal fatigue cracks. In addition to annotating the defects, two types of masks were manually marked on each image. The weld areas were masked for later use in the virtual flaw data augmentation step and evaluation. Drilled holes in image quality indicators (IQIs) looked very similar to pores and lowered performance slightly when included as non-defective examples. Thus, they were masked to allow them to be excluded from the training data.

For the initial development phase, the raw images were randomly divided into training (60%), validation (20%) and test (20%) data sets. The number of original cracks was insufficient for both training and validation, thus all images with original cracks were placed in the validation or test sets, and the training set contained all of the manufactured cracks, i.e. only virtual flaws were used in the case of cracks. The training and validation sets were later combined for use in cross-validation to compare the three different augmentation methods. The test set was reserved for a final evaluation.

2.3 Evaluating the significance of virtual flaw data augmentation

We compared 3 different data augmentation strategies: standard (using random shear, rotation, crop and resize, flips, noise, brightness and contrast), pure virtual, using virtual flaws only, and combined, an even sampling of the first two with 50% of each. The processing pipelines resulting in the three augmented data sets are presented in Fig. 2.

For standard augmentation, 512×512 pixel patches were randomly extracted from the original images, half containing defects and half being clean. Because the linearity of features, like weld edges or cracks, have significance in our case, we used affine transformations that preserve straight lines as opposed to the random elastic deformations used by Ronneberger et al. [42].

For virtual flaw augmentation, we used a method similar to Virkkunen et al. [50], modified for radiography data. The annotated flaws were extracted from background. The extracted flaws were augmented with affine transformations, random noise and flips, and then re-introduced onto non-defected areas on the welds by utilizing the manually marked weld masks. Again, size 512×512 pixels image patches were randomly sampled from the training images, with 50% flawed and 50% non-flawed examples. To
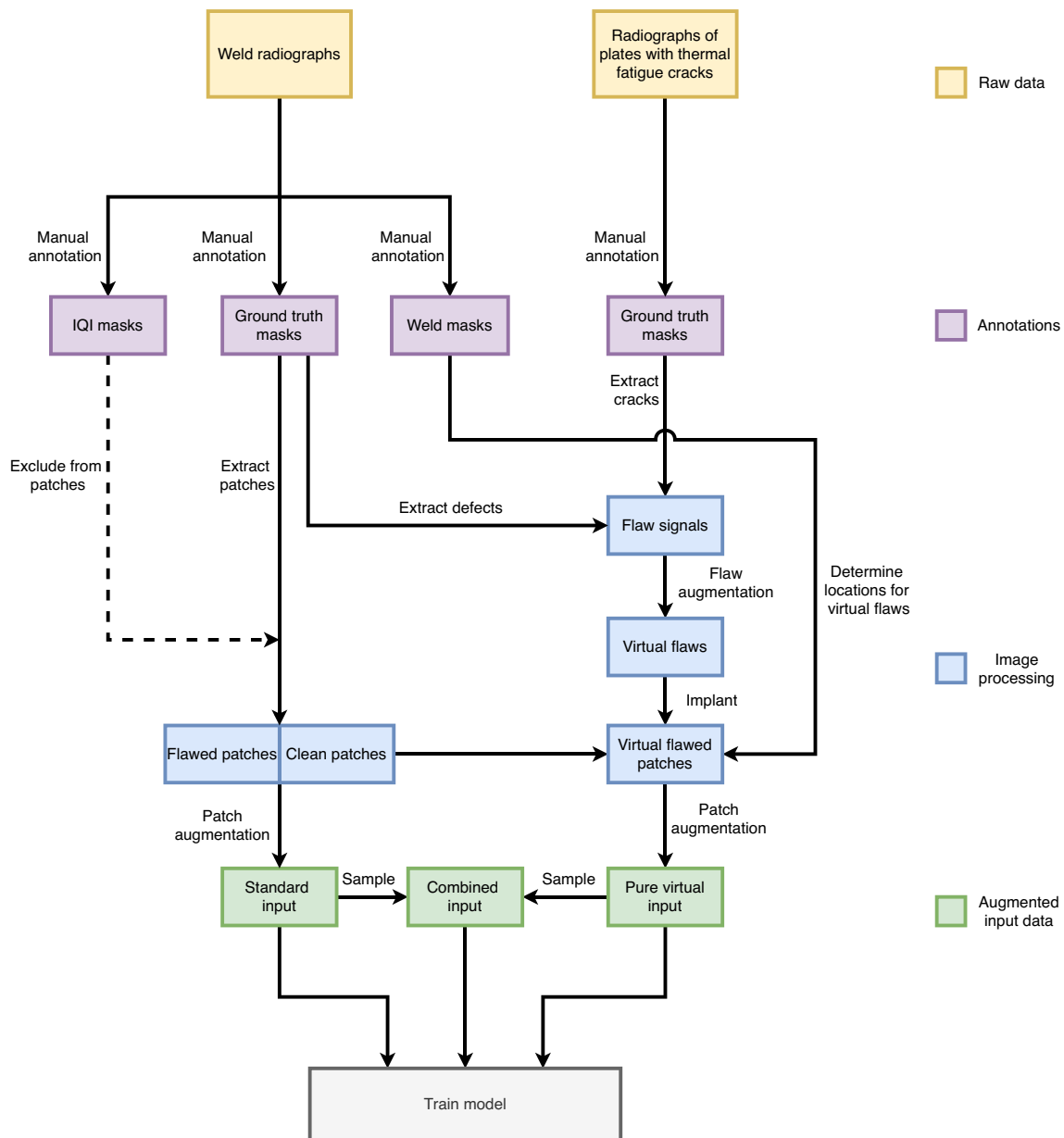
Fig. 2 Data processing steps resulting in three different augmented inputs: standard, pure virtual, and combined. The processing consists of four stages: collecting raw data, making annotations, image processing, and sampling the standard and pure virtual data sets for a combined input

represent clustered defects commonly found in welds, up to five virtual flaws were randomly implanted into each flawed example, but with fewer (1–2) being most common.

Extracted thermal fatigue crack signals were added in the virtual flaw augmentation. To compensate for the small number of cracks, the crack signals were over-represented in comparison to their number in the original data set, so that they made up 5% of all the implanted flaws. The resulting implanted cracks have lower natural variation than the pores, but the affine transformations make up a fairly representative distribution of flaws for the task, similarly to the defects in a simulation approach by Mery [33].

Previous studies by Virkkunen et al. [50] and Koskinen et al. [27] using virtual flaw augmentation on UT data had a limited amount of real flaws in their data sets, so a pure virtual flaw approach was used. However, the material in this study provides a larger representation of defect locations in its original form, without possibility for artefacts or skewed distribution that may be present in the virtual flaw process. Thus, we studied if combining standard augmentation and virtual flaw augmentation can improve results by creating a 50/50 sampled data set from the standard and pure virtual augmentation. Examples of standard and virtual flaw augmentation are shown in Fig. 3.
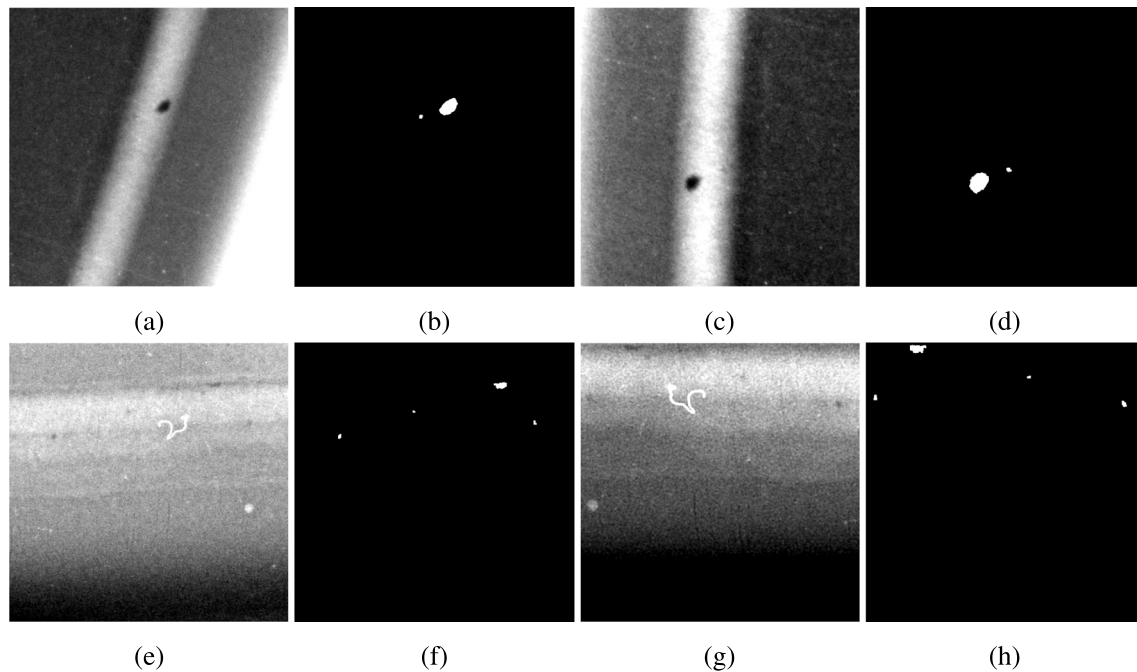
Fig. 3 Patches of training data and their labels for virtual flaw and standard augmentation. First row: Virtual flaw. (a) After flaw implanting. (b) Corresponding ground truth mask. (c) After further augmentation. (d) Corresponding ground truth mask. Second row: Standard augmentation. (e) Unedited patch. (f) Corresponding ground truth mask. (g) After augmentation. (h) Corresponding ground truth mask
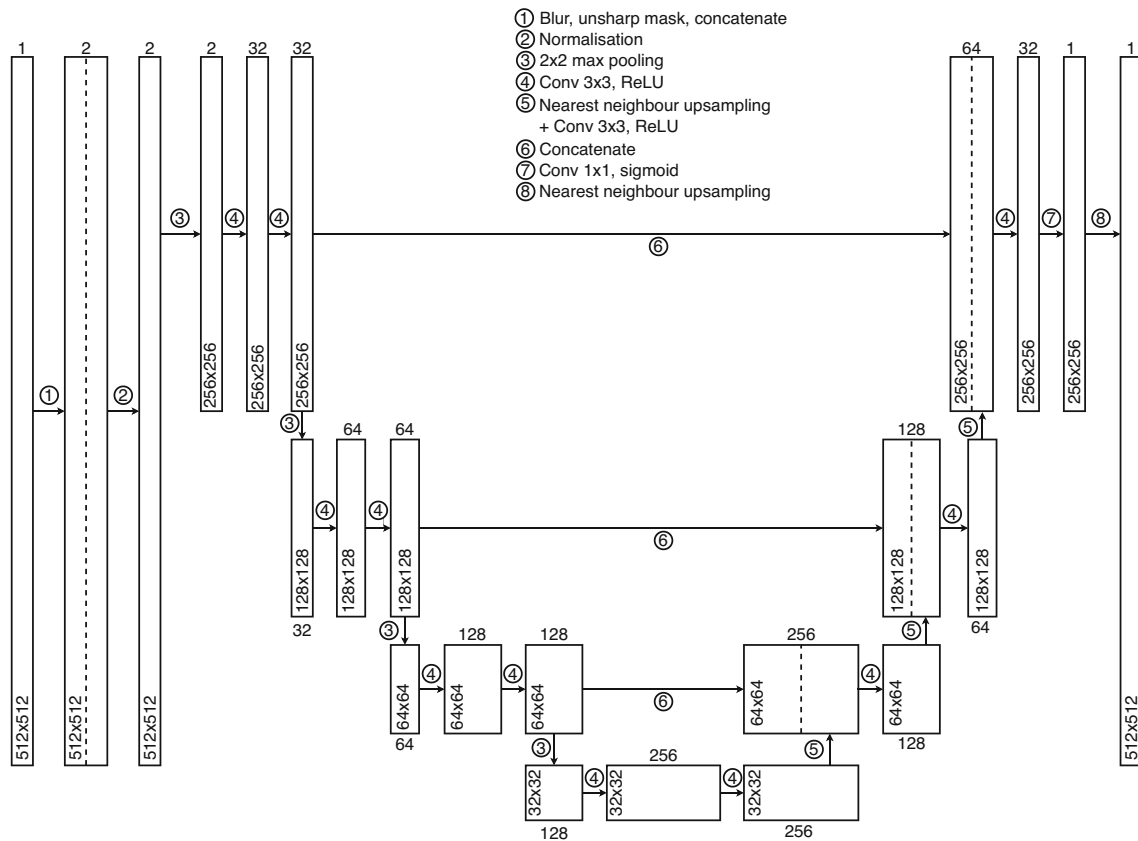
2.4 Model architecture

We chose a semantic segmentation approach based on the following considerations. Defects that are intersecting or are close to each other are usually treated as a single large indication in terms of inspection acceptance, thus it is not necessary to accurately separate defects. In cases like this, semantic segmentation architectures can be used instead of instance segmentation without loss of benefit for the use case. This simplifies the annotation process, because overlapping or adjacent indications do not have to be separated. Object detection networks (which output bounding boxes), on the other hand, give less information of the shape of defect, which can be problematic for example in the case of a diagonally placed crack, that would produce an approximately square bounding box. We later used post-processing steps dependent on the linearity of the defect, so the per-pixel annotation was necessary.

We preprocessed the data by unsharp masking to highlight flaws, similarly to what is used by human inspectors, and concatenated the blurred and sharpened images into a 2-channel input image to preserve the original information. We downsampled the input from 512×512 to 256×256 pixels to reduce computational costs on the very high resolution input images. This was done by first inverting the images, so that the defects were light (large values), and then max pooling. Finally, we normalized the

images batch-wise by subtracting the mean and dividing by the standard deviation.

A modified U-net by Ronneberger et al. [42] was chosen due to promising performance in similar segmentation tasks, a simple implementation, and its speed at inference time. A schematic of the architecture used in this work is shown in Fig. 4. It is a fully convolutional encoder-decoder architecture with skip connections between matching encoder and decoder stages. The inputs are size 256×256 pixels image patches, and the output is a pixel-wise classification mask with 0-valued pixels denoting background and 1-valued pixels indicating defect. The model architecture is shown in Fig. 4. The differences to the original U-net are as follows. We used less filters to make training and inference faster, since no loss of accuracy was found from reducing the model size. We changed upwards convolution to upsampling + convolution for compatibility with an optimized TensorRT version of the model used in deployment. We used padding in the convolutions to keep the height and width constant between pooling layers. Finally, we used a sigmoid activation to produce binary classification (defect vs. background) instead of 3-class (foreground, background, edge) output since we did not require instance-aware segmentation (separating overlapping flaws).

To account for the unbalance between the number of background and flaw pixels, we used a weighted binary

Fig. 4 A schematic of the modified U-net model architecture used for detecting flaws in radiographs. The rectangles represent a single example as it propagated trough the network. Their dimensions reflect the dimensions of the data. The value on top or bottom of each rectangle shows the depth of the processed tensor (or image), while the values inside each rectangle are the width and height. The input image, on the left, is a 512×512 pixel monochromatic image. The image is first pre-processed by unsharp masking, normalizing and max pooling to half dimensions, followed by two sets of 3×3 convolutions with 32 filters each. The max pool and two convolutions block is repeated two times, doubling the filters each time. Two 3×3 convolutions are done between the encoder and decoder stages as shown in the bottom middle section. This is followed by 3 upsampling blocks: upsampling, convolution, concatenation from the equally sized encoder stage, and another convolution. Finally, a 1×1 convolution with sigmoid activation produces the binary segmentation masks, which are upsampled to original size. In total, the model has 1.7 million trainable parameters

cross-entropy that weighs losses in flawed pixels higher than background pixels similarly to Ronneberger et al. [42]. The weight was a tuned hyperparameter, we found 3 to work best. The model was trained using the Adam optimizer with a batch size of 32. Learning rate was halved every 2500 steps if no improvement to validation loss occurred. The weights producing the lowest validation loss during training were saved.

2.5 Inference and post-processing of model output

To facilitate the memory-intensive computations, the model takes much smaller image patches as input than what the original radiographs are. For inference on these large images, a sliding window was first used to divide each image into patches with some overlap to increase robustness. As in the inference approach by Ronneberger et al. [42], the borders are reflected to better find defects

in the edges of the image. After inference, the resulting masks are joined back into a full-size annotation mask. Overlapping regions are joined by an OR-operation, that is, an indication in either overlapping patch is added to the full-sized mask.

After generating full-sized masks for input images, further post-processing was done to apply acceptance rules based on size, shape, and proximity. The classification masks were post-processed as follows. The individual indicated areas were measured by fitting a circle around each masked area. The image pixel to millimeter scaling information was used to give each indication's diameter in millimeters. Indications very close to each other were merged, according to acceptance rules stating that defects near enough to each other are interpreted as a single large defect. Linearly shaped defects were classified by fitting a rectangle around each indication and flagging those with a narrow aspect ratio. Flaw clusters were detected

by proximity and porosity chains were detected from clusters by using the same linearity test as for individual defects.

After applying acceptance rules, the resulting annotations were visualized by circles. Defects and clusters that were classified as critical, either due to size or crack-like morphology, were marked by white circles. Acceptable defects were marked by black circles. The smallest individual indications under a given detectability limit from the inspection were not marked.

2.6 Evaluation

The standard validation metrics in deep learning and NDE emphasize slightly different areas. To receive acceptance in industry applications, the validation of an automated system should fit NDE practices. A POD is often used to measure the capability of the whole NDE system, including equipment and analysis. The POD evaluation in this paper is concerned with detection from available radiographs.

Following considerations in Section 1.7, we interpreted any overlap as a hit when calculating the POD. We used the POD and 95% confidence bounds to determine the $a_{90/95}$ flaw size, which is the size of defect that is detected with 90% probability with 95% confidence. If significantly large defects were present in the material, another metric would be required, however, all flaws in the data set were small.

We defined false call rate as the number of false calls per unit length of weld. The lengths of the welds were estimated by skeletonizing the manually labeled masks of welds with the thinning algorithm by Guo and Hall [15]. The number of false calls is the number of predicted regions with no overlapping ground truth. We observed that a significant portion of the false calls occurred outside of the weld area. When used as an inspector's aid, the user is only going to consider responses that appear on the weld. To reflect performance in this primary use case, we also measured the false call rate restricted to the weld area for which used the manually annotated weld masks to filter out false calls outside the ROIs. In a later study, we plan to automatically segment the weld areas to exclude indications outside the ROI.

We measured the sizing performance, or accuracy, of the system as follows. For each hit, we used an enclosing circle fit to measure the radius of the true and predicted indication. If more than one prediction overlapped with the true defect, we chose the one with the largest IoU. We calculated the average absolute error in millimeters.

Using the aforementioned metrics, a fivefold cross-validation with seven subsets of data was performed as follows. For each of the five runs in the cross-validation, a holdout set of one-fifth of the data was reserved and the rest was used for training and validation. Within each run, seven subsets of the training data with decreasing size were used: 100%, 75%, 50%, 25%, 10%, 5% and finally 1.5%. The sizes of the data sets are presented in Table 1. The smallest data set contained about 225 unique 512×512 pixel patches of data and 14 defects on average, representing an extremely limited material. The holdout sets consisted of 35 raw images, divided as described in Section 2.5 to form about 35000 patches (without overlap, about 7875 unique patches sized 512×512 pixels). The training set was randomly sampled from the available training data according to each subset size. A validation data set with 20% of the remaining training data was used to save the best performing model during training. The folds had slightly varying amounts of defects, because data was sampled as entire images, each with a different number of defects. The compared augmentation methods used the same subsets of raw data.

Each training set was used to train 3 models: The first using standard augmentation, the second using purely virtual flaws, and the third using an even sampling of standard and virtual flaw augmentation. Each model was trained for 15000 steps with batch size 32, using the Adam optimizer with initial learning rate 0.0005. The learning rate was halved every 2500 steps if no improvement occurred on the validation set. The training was carried out on an Nvidia RTX 3090 GPU, and took about 1 hour per model.

Table 1 The size of train, validation and test data for each run in the cross-validation for decreasing data set size, ranging from 100 to 1.5% of data used: the number of training and validation images sized 7750×7750 pixels on average per fold, the approximate number of unique 512×512 pixels patches in the images, and smallest number of defects out of the folds in each fraction

Subset	Images		Unique patches		Defects (smallest fold)		
	Train	Validation	Train	Validation	Train	Validation	Test
100%	112	28	25200	6300	1711	399	497
75%	84	21	18900	4725	1276	285	516
50%	56	14	12600	3150	783	178	499
25%	28	7	6300	1575	347	69	491
10%	11	3	2475	675	173	48	472
5%	6	1	1350	225	73	9	476
1.5%	1	1	225	225	7	3	432

2.7 Field evaluation

After quantitative evaluation (Section 2.6), a model trained using combined augmentation was deployed in a test setting as a part of an inspection pipeline. A standalone edge computing unit without internet connection was used to facilitate use in a high security environment. The model was converted to TensorRT and integrated into a software which generated annotations from radiograph input. The input and output data formats were set to work with the users' viewing software. Visualizations based on the segmentation and acceptance rules were evaluated by personnel working with the radiographic inspection by comparing their analysis with annotations provided by the model. Qualitative results were collected by discussing with the users. The users were asked to describe the system's reliability, transparency, ease of use and benefit to the inspection procedure. The hardware of the prototype was less powerful than the unit planned for deployment, and the system exhibited some incompatibilities with the users' software. The field study had limited participants (three) and was thus indicative. In the future, we plan to address the hardware and software issues to conduct more comprehensive field testing.

3 Results

3.1 Performance and POD

The models achieved a good inference speed. A size 7900×8300 pixels image, divided into 960 patches, takes about 6 seconds (6.3 ms per patch) to annotate on an Nvidia GTX 3090 graphics card, or 15 seconds on an Nvidia Jetson AGX Xavier running a TensorRT-converted model.

POD curves for the combined augmentation model trained on 100% of the available training data are shown in Fig. 5. Five curves are plotted, one for each trained and validated model in the cross-validation. All POD curves are shown in Appendix A. Due to the large test set size, the lower 95% confidence bound (dashed line) is very close to the curves. The resulting $a_{90/95}$ is small, indicating that the model is sufficiently sensitive.

3.2 Comparison of data augmentation methods

Four metrics were used to compare standard, pure virtual and combined augmentation. The worst case results from cross-validation were presented for each: $a_{90/95}$, sizing error, false call rate on weld area, and false call rate on the entire image.

Figure 6a shows results for $a_{90/95}$. Combined augmentation achieved best results for all fractions of data,
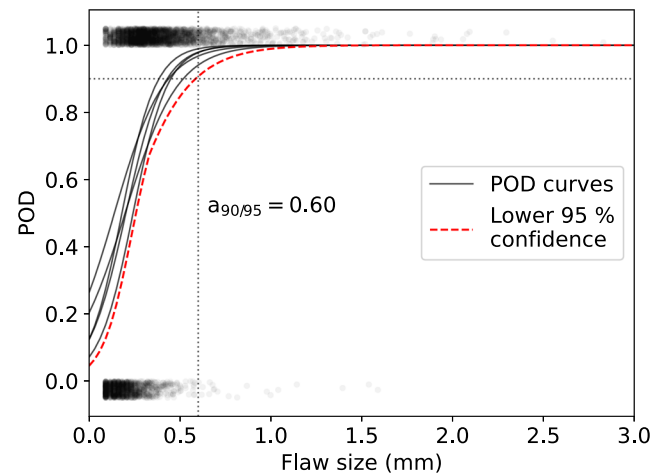

Fig. 5 POD curves for a deep learning model trained using the combined (standard and virtual flaw) augmentation and 100% of available training data. As a result of 5-fold cross-validation, 5 separate POD curves are drawn, each for a separately trained and validated model. Hits are marked by black dots on the top of the plot, and misses on the bottom. The dashed curve is calculated by taking the minimum of the lower 95% confidence bounds of each POD curve. The intersection of the lower confidence bound with POD = 0.9, marked by the dotted horizontal and vertical lines, is the worst case $a_{90/95}$

with significant differences to standard and pure virtual augmentation at small data set sizes. For combined augmentation, the $a_{90/95}$ remained small even at very small subsets. Virtual flaw augmentation and conventional augmentation performed roughly equally well with the larger data sets, indicating that the material was sufficient to largely capture the important features in the inspection regardless of augmentation method — performance was likely more limited by the obscure line between defect and non-defect at small indication sizes. With decreasing data set size, both methods with virtual flaw augmentation significantly outperformed regular augmentation. This is likely due to the insufficient variability in flaw locations and sizes with the conventional method.

Figure 6b presents the sizing error results. Differences are quite small, with standard augmentation and combined augmentation giving better results overall. The differences between the methods were smaller than for $a_{90/95}$ and the sizing errors were generally small, indicating that once a signal is correctly classified as a defect, its accurate segmentation is an easier task for the deep learning model.

The results for false call rate on weld area are shown in Fig. 6c. The rates were quite close to each other at about 1–2 false calls per 10 cm of weld, with no significant increase until 5% or less of the training data was used. At 5%, the combined method remained at the same level while pure virtual and standard augmentation made much

Fig. 6 Four NDE evaluation metrics vs. fraction of data for the three augmentation methods: standard, pure virtual and combined augmentation. Worst case results from fivefold cross-validation are displayed. Lower is better for all metrics. (a) $a_{90/95}$. (b) Sizing error. (c) False call rate on weld area. (d) False call rate on image

(a)

(b)

(c)

(d)
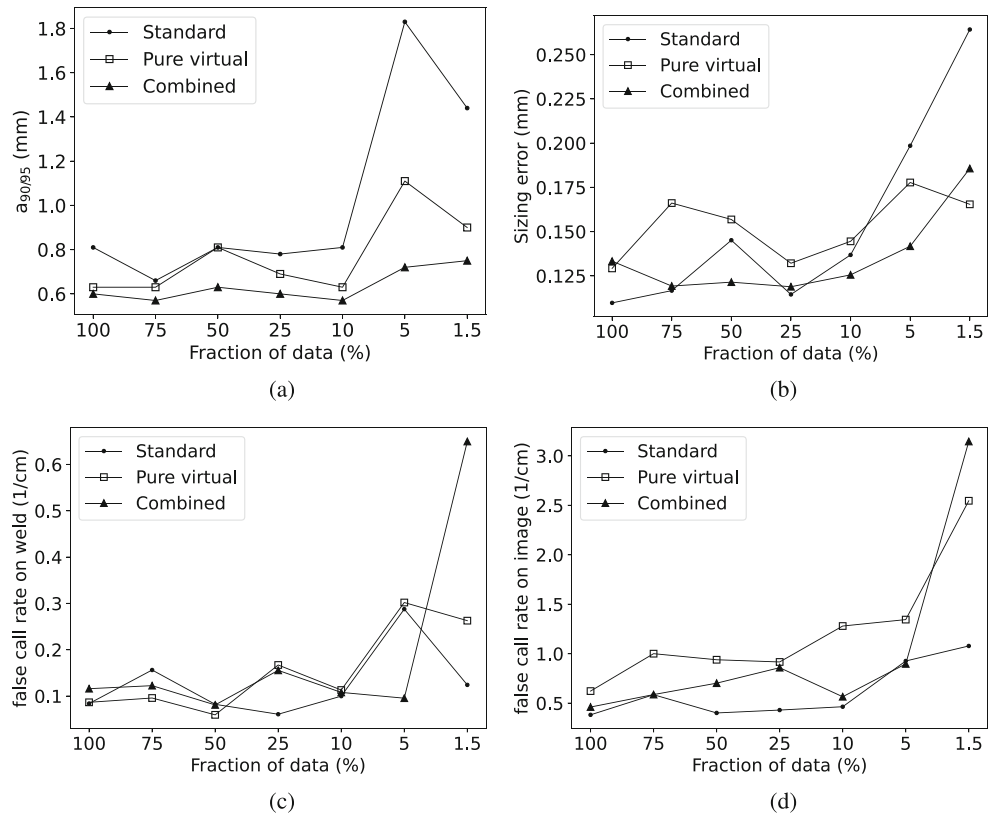
more false calls. At 1.5%, the combined method begun to show excessive false calls while the virtual and standard augmentation made less.

False call rates on entire images (also outside of the weld area) is shown in Fig. 6d. The rates were significantly higher than for weld areas, revealing that most false calls

Fig. 7 Segmentation masks for a patch of test data on models trained with 5% of the data set, illustrating performance differences in segmenting pore clusters. (a) An input image with clustered porosity on and next to the weld. (b) Manually annotated ground truth. (c) Standard augmentation. The predicted segmentation is missing some larger pores, and finding others seemingly randomly. (d) Virtual flaw augmentation. The mask is very close to ground truth, with a small miss in the middle. (e) Combined augmentation with a prediction nearly identical to ground truth

(a) Input image

(b) Ground truth

(c) Standard

(d) Pure virtual

(e) Combined
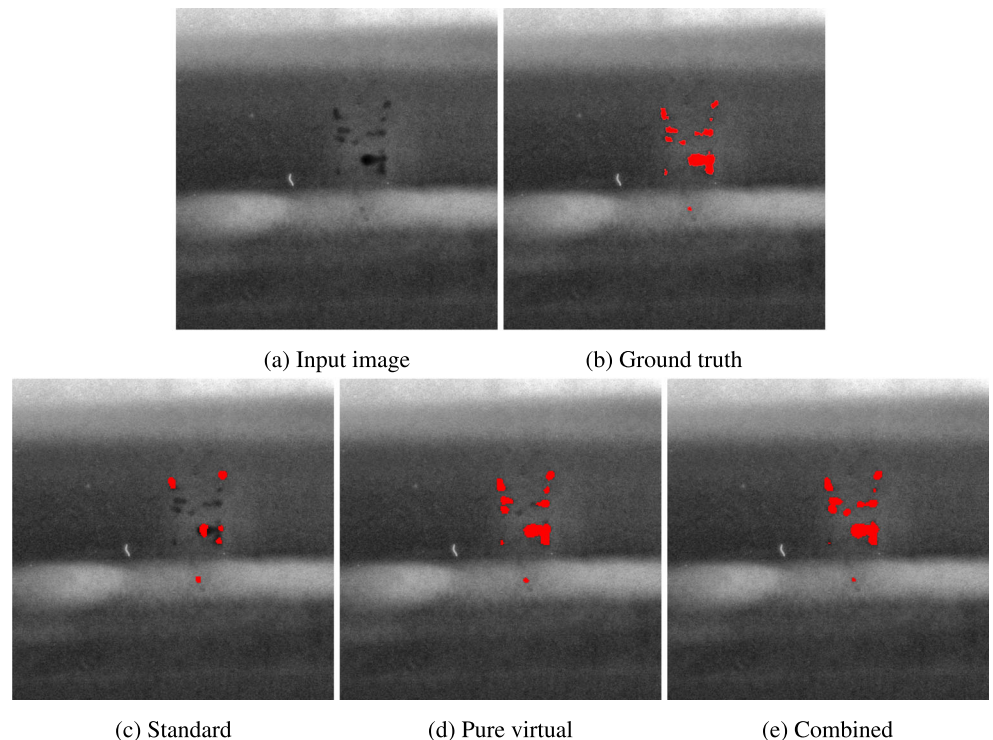
Fig. 8 Segmentation masks for a patch of test data on models trained with 5% of the data set, illustrating performance differences in segmenting large cavities. (a) An input image with an unusually large cavity. (b) Manually annotated ground truth. (c) Standard augmentation. The small, insignificant pores are found, while the large cavity is missed. (d) Virtual flaw augmentation, also missing the large cavity. (e) Combined augmentation. The large cavity is correctly segmented

(a) Input image   (b) Ground truth

(c) Standard   (d) Pure virtual   (e) Combined
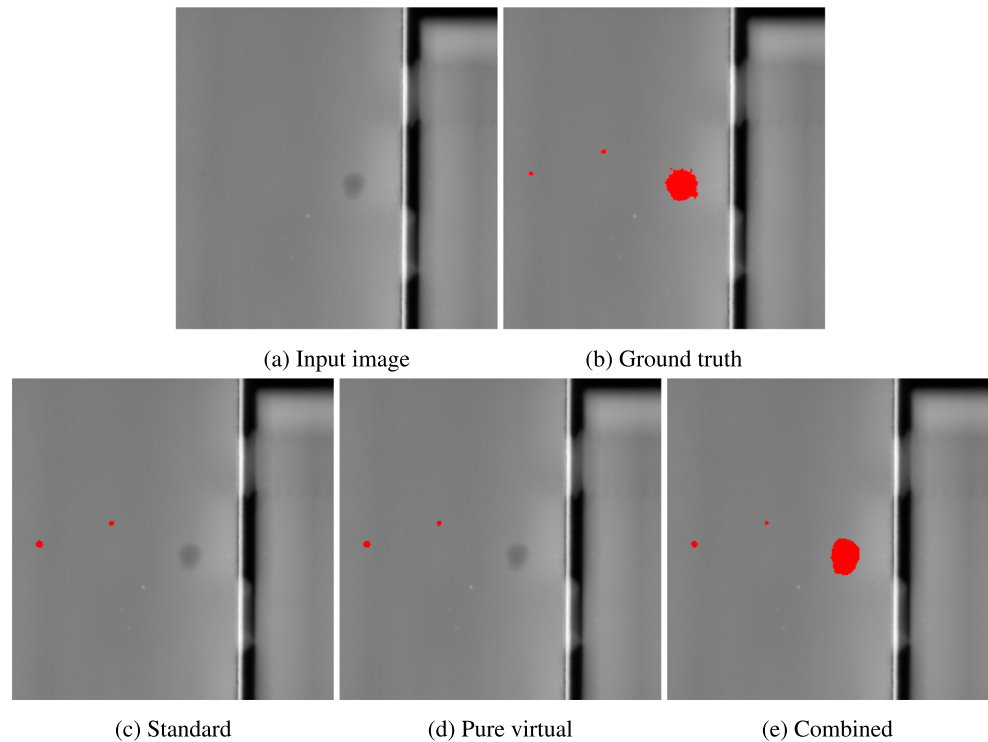
occurred outside ROIs. Pure virtual flaw augmentation performed worse than the standard and combined methods. The standard augmentation made least false calls overall with combined augmentation achieving matching results for 75% of the data and better results for 5%.

3.3 Qualitative results

Segmentation examples from the models are presented in Figs. 7, 8, 9, 10 and 11. To highlight differences between the methods, we show predicted annotations generated by

Fig. 9 Segmentation masks for a patch of test data on models trained with 5% of the data set, illustrating performance differences in sizing. (a) An input image with a medium-sized pore. (b) Manually annotated ground truth. (c) Standard augmentation. The defect is found, but sized too small. (d) Virtual flaw augmentation, giving an accurate sizing. (e) Combined augmentation, giving an accurate sizing

(a) Input image   (b) Ground truth

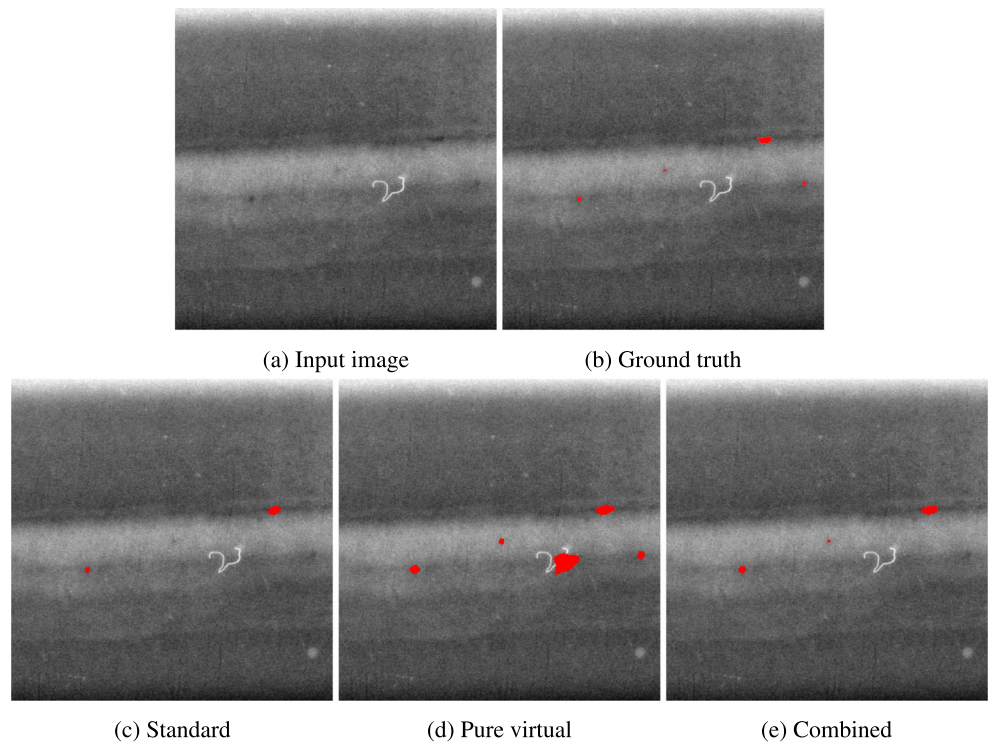(c) Standard   (d) Pure virtual   (e) Combined
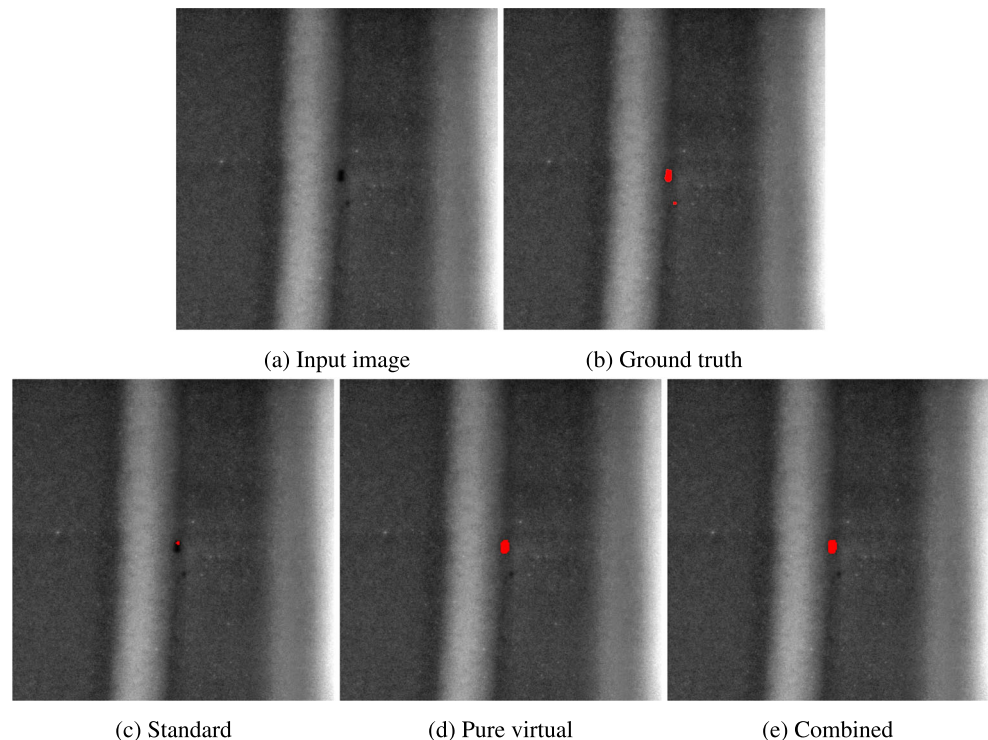
Fig. 10 Segmentation masks for a patch of test data on models trained with 5% of the data set, illustrating performance differences in false calls. (a) An input image with two pores. (b) Manually annotated ground truth. (c) Standard augmentation. The defects are found, but two false calls are made with no clear explanation. (d) Virtual flaw augmentation. The defects are found without false calls. (e) Combined augmentation. The defects are found without false calls

(a) Input image   (b) Ground truth

(c) Standard   (d) Pure virtual   (e) Combined
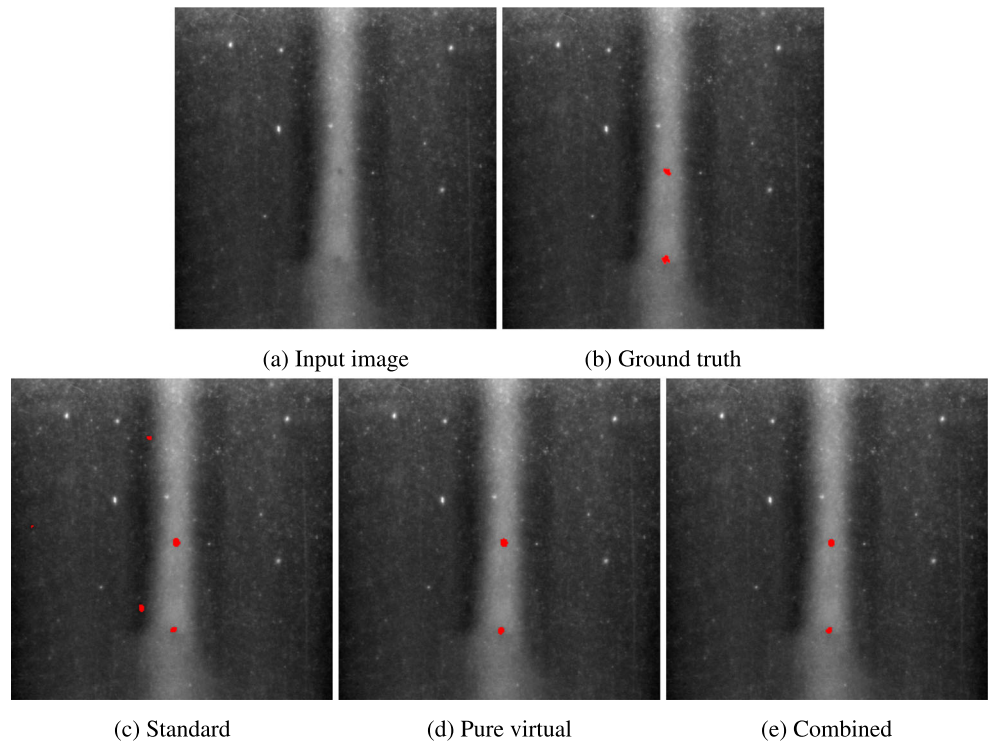
models trained on a 5% fraction of the training data, where the performances have clearly diverged. A random sample of predictions at different data set sizes is provided in Appendix C.

Figure 7 shows results for an input image with clustered porosity. Standard augmentation performed poorly in segmenting a pore cluster. Figure 8 displays predictions for an input with an unusually large pore. The combined

Fig. 11 Segmentation masks for a patch of test data on models trained with 5% of the data set, illustrating performance differences in false calls. (a) An input image with small pores and a foreign particle, unimportant for the inspection. (b) Manually annotated ground truth. Three very small indications are included. (c) Standard augmentation, no false calls. (d) Virtual flaw augmentation. The particle is falsely segmented as a large defect. (e) Combined augmentation. No false calls

(a) Input image   (b) Ground truth

(c) Standard   (d) Pure virtual   (e) Combined

Fig. 12 Segmentation masks for a patch of test data on models trained with 100% of the data set, illustrating performance differences in finding linear defects. The virtual and standard augmentation data sets contained virtual flaw cracks that were extracted from data outside of the inspection case. (a) An input image with a large crack. (b) Manually annotated ground truth. (c) Standard augmentation. The crack is missed. (d) Virtual flaw augmentation. The crack is correctly segmented. (e) Combined augmentation. The crack is correctly segmented

(a) Input image   (b) Ground truth

(c) Standard   (d) Pure virtual   (e) Combined
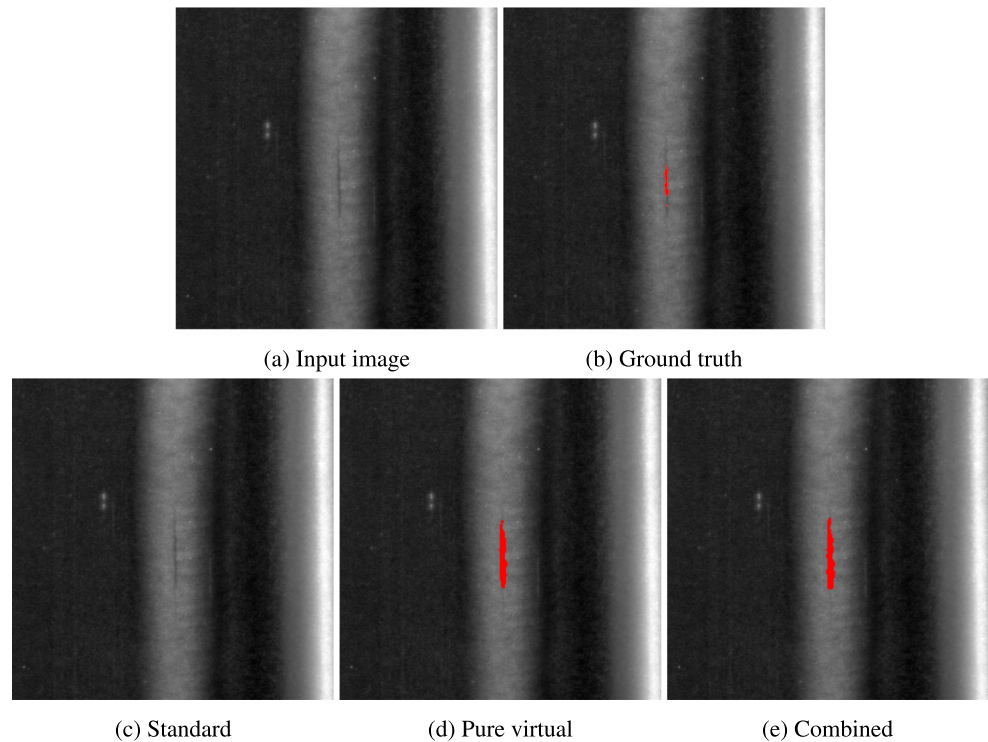
augmentation successfully segmented the defect, while the pure virtual and standard methods missed it. Figure 9 illustrates the differences of the methods in terms of sizing capability. The methods involving virtual flaws have retained their sizing capability with small training data, while the standard augmentation significantly underestimated the pore size. Figures 10 and 11 show two cases of false calls. In Fig. 10, some unexplained false calls were made by the standard augmentation. In Fig. 11, a foreign particle caused a false call for the pure virtual flaw method.

Finally, we show an example of capability in finding linear defects in Fig. 12, this time for models trained on 100% of the available data, since the differences are clear across all fractions. Both methods involving virtual flaws segmented the crack correctly, while the standard method missed it.

3.4 Field experiment results

The model prototype was successfully deployed in a high security environment. An example of the visualizations used in the field experiment is shown in Fig. 13. Indications were found to match expectations fairly well. Misses were found on the low, acceptable end of the flaw size, some of these were indications left unmarked due to acceptance criteria. Users found the false calls outside of the weld

Fig. 13 Visualizations used in the deployed system. (a) Two acceptable pores (black circles). (b) An unacceptable pore group (merged white circles), with an annotation (0,3/2) indicating a group of two pores with largest diameter 0.3 mm

(a)   (b)
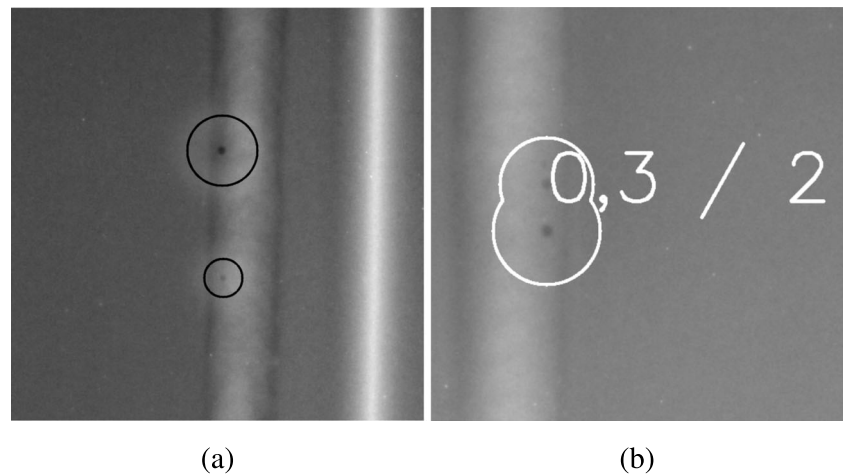
area not to affect the inspection directly, but thought they could be potentially distracting over a longer period of use. The annotations provided by the model were deemed easy to understand. When indications were not present, due to misses or pruning small indications, the behaviour was found more difficult to explain. In general, understanding the way the model operated was important for the users. The edge computing unit and the generated visualizations were found to be easy to use. Inference speed strongly affected the user experience. Overall, users saw potential for the system to be used as an inspector's aid.

4 Discussion

The $a_{90/95}$ of 0.6 mm for combined augmentation (Fig. 5) indicates high reliability for the inspection case. It more sensitive than the NASA-STD-5009B [37] reference value for minimum detectable crack sizes, although this is indicative since the material mostly consists of pores. MIL-HDBK-1823A [1] recommends a minimum of 60 flaws for a hit/miss POD curve, which results in significant differences between the POD curve and confidence bounds. Due to the much larger test set sample size in this study (about 500 on average), the lower 95% confidence bound (dashed line) is close to the POD curves. Considering the patch size (512×512 pixels), the material was large in comparison to other weld data sets [23, 35].

Adding virtual flaws improved detection at smaller data sets. Figure 7 shows how standard augmentation segmented a cluster poorly, likely due to the lack of flaw groups in the training data. Adding several virtual flaws to some examples represented clusters well. In terms of $a_{90/95}$, combined augmentation performed best on all fractions. Even when using only one image for generating training patches, the combined data augmentation achieved a good $a_{90/95}$ result. With pure virtual flaw augmentation, used by Koskinen et al. [27], there can be a loss of some subtle features related to the combination of location and signal, not perfectly captured by virtual flaw, which is then alleviated by mixing defects in their original locations and new locations. Moreover, the distribution of flaw sizes and shapes is slightly skewed from the original, the effect of which is reduced by having half of the data set follow the original distribution. In Fig. 8, for example, the pure virtual flaw augmentation model missed a large cavity, which the combined augmentation model found. The $a_{90/95}$ remained low for surprisingly small amounts of data, indicating that the proposed system can be scaled to other inspection cases with a moderate amount of manual annotation.

Performance differences in finding linear defects (Fig. 12) indicate firstly that the linear defects are different enough from pores, that if not included in the training

data, they may be missed. Secondly, using the virtual flaw to extract suitable flaw signals from other components than the ones being inspected was a successful strategy for covering a defect type which was scarce in the primary data. This is a simpler approach in comparison to simulation methods [13, 33] that require each defect type to be modeled in a representative way. Koskinen et al. [27] also found simulation to yield limited generalization. In the case of cracks, the combined method did not differ from pure virtual, since no cracks from the original material were present in the training data set, but they were rather used for validation.

Pure virtual augmentation sized the defects less accurately than the other two methods. The ground truth masks for the virtual flaws, generated during the implanting, likely caused a small discrepancy in comparison to the regular annotations. Again, mixing regular and virtual flaw augmentation resolved this. At smaller data set sizes, conventional augmentation started to perform significantly worse, in correlation with the much higher $a_{90/95}$ values, indicating that the model failed to find a reasonable fit. An example of deteriorated sizing performance for the conventional augmentation model is shown in Fig. 9, where a medium-sized pore was significantly undersized.

The manual annotations for the smallest, acceptable weld defects display some differences. A region of uncertainty exists where there is no clear line between defect and non-defect like noise or geometry, even for a human inspector. This is the case with most NDE applications: deciding if a signal is an imperfection becomes increasingly difficult near the limits of what is detectable by the imaging method. Figure 9, for instance, shows a very small pore annotated in the ground truth, but not indicated by any of the machine learning models. It is unclear whether the model or manual annotation is more correct, since an indication that small could also be just noise or geometry. A similar effect was observed by Mery [33]. This noise in the labeling makes the models exhibit poor separation at edge cases in the small flaw range (which is not of interest for the inspection), but notably, this does not hinder performance on the larger, unacceptable defects. At small defect sizes, the labeling noise also limits the accuracy of the calculated POD. In medical radiography data sets, like CheXpert [22], committees of experts have been used for improved ground truth labeling, as well as uncertainty labels to reflect difficult to judge cases. This is more resource intensive, but potentially of interest in future studies. In the context of NDE, validation via destructive testing like macrography could provide more accurate ground truths, but is infeasible due to the large number and small size of the defects.

Looking at $a_{90/95}$ and false call rate simultaneously, it can be seen that the methods responded to insufficient data differently: while the standard augmentation started missing more flaws, the combined method made excessive false

calls. The pure virtual flaw was in the middle for both of these metrics. The $a_{90/95}$ is the more crucial metric out of these two, indicating a safer failure mode occurred for the methods using virtual flaws. Much of the false call rate on the weld area can be attributed to the annotation uncertainty at the small acceptable sizes.

Using pure virtual implanted flaws caused a higher tendency for false calls outside of the ROI. Combined augmentation reduced this significantly, although not completely. Excessive false calls outside of weld areas are not problematic for use as a human aid tool, but using more automatic systems or collecting statistics would require them to be reduced to lower levels. In the future, we plan to address this by automatically segmenting the welds to prune the indications outside of the regions of interest.

The results were reported in standard NDE metrics. Other research in the application area [11, 13, 33] report deep learning oriented metrics like mean average precision (mAP) or receiver operating characteristic (ROC) curves using an IoU threshold, which makes industry adoption and the comparison of NDE performances more difficult. POD gives the information of flaw size vs. detection, which is important for NDE. Moreover, POD and false call rate in NDE separate the analysis of detection and false calls, while the common segmentation metrics combine them.

The U-net architecture by Ronneberger et al. [42] was found to be flexible for small modifications. Performance was found to be strongly driven by data set qualities like number and type of defects, labeling noise and augmentation methods, which indicates that other architectures like FPN by Lin et al. [30] or mask R-CNN by He et al. [17] are not likely to make significant improvements to performance. A comparison across multiple architectures similarly to Mery and Arteta [34] or Mery [33] is of interest in future research.

The field experiment gave indications of fairly good agreement with human operators, with some issues related to small edge-case defects and false calls outside of the ROI. The proposed method of deployment and user interface were found to be easy to use and well suited for the application, and compatibility with existing industry software made the deployment simple. The importance of system transparency was highlighted: for example, leaving the smallest defects unmarked likely reduced the perceived robustness of the model, since the information of whether defects were missed or only left unmarked due to small size was not easily available.

To summarize, deep learning-based segmentation is a feasible approach for automating industrial radiography inspections for a challenging weld case, capable of fulfilling strict requirements in the aerospace field. The proposed model indicated good sensitivity in comparison to the NASA-STD-5009B [37] reference value. We found a

combined approach of virtual flaw augmentation and standard data augmentation gave a significant performance increase in the most important metric, the $a_{90/95}$, especially at smaller subsets of data. The combined method also performed sufficiently well in sizing and false calls, making it the overall best method out of the three, an improvement over pure virtual flaw augmentation. Good performance on cracks was achieved by using virtual flaws, which is a more straightforward strategy than simulation methods for highly varying defect types. By comparing with qualitative results, the adopted NDE metrics were found to represent performance well, and they facilitate use in industry. The field experiment gave positive indications for future deployment as an assisting tool for inspectors.

5 Conclusion

We developed a deep learning-based system to automatically detect, segment and rate the severity of flaws in welds in aerospace components. Standard metrics in NDE were adopted for a deep learning approach, and three augmentation methods were compared: standard (using random shear, rotation, crop and resize, flips, noise, brightness and contrast), pure virtual flaw, and a combination of the two. A field experiment was conducted in a real industry setting. The best method using combined augmentation achieved high sensitivity, accurate sizing and acceptable false call rate, sufficient for strict aerospace weld requirements. Using virtual flaws was found to increase the detection capability of the model, and combining original and virtual flaw data largely alleviated possible implanting-related issues, like artefacts, annotation noise or skewed flaw distribution. Small, acceptable defects had inconsistencies in annotations when close to undetectable, contributing to false call rate and small misses, but not adversely impacting performance on the more critical, large defects. The adoption of industry-standard metrics advanced deep learning-based automation methods towards commercial use in weld NDE. We demonstrated that deep learning-based weld defect detection can reach high performance and be deployed in real industry environments.

Appendix A. POD curves

The POD curves of all models trained in the experiments are shown in Figs. 14, 15, and 16. The curves are organized in the following way. For each subset of data, ranging from 100 down to 1.5%, three figures are presented: one for standard augmentation, virtual flaw augmentation, and combined augmentation respectively.

The curves become more shallow at lower fractions of data, resulting in higher $a_{90/95}$ values and thus a poorer performance. The spread of the curves also increases for smaller fractions, indicating that the individual cross-validation results start to differ significantly depending on the random sample of data used for training, validation and testing. The standard augmentation curves (first row in Figs. 14, 15 and 16) have the most spread, while the combined augmentation curves (third row) have the least. This suggests that adding virtual flaws increased robustness.

The curves, calculated following the standard POD approach, do not show 0% probability of detection for zero-sized flaws. This is a known limitation in current POD analysis methods, which stems from an insufficient separation of hits and misses in the range of very small flaws. The problem is sometimes alleviated by artificially adding near-zero misses in the plots to force wanted behaviour. See Virkkunen [48] for more discussion on this problem and potential remedies.
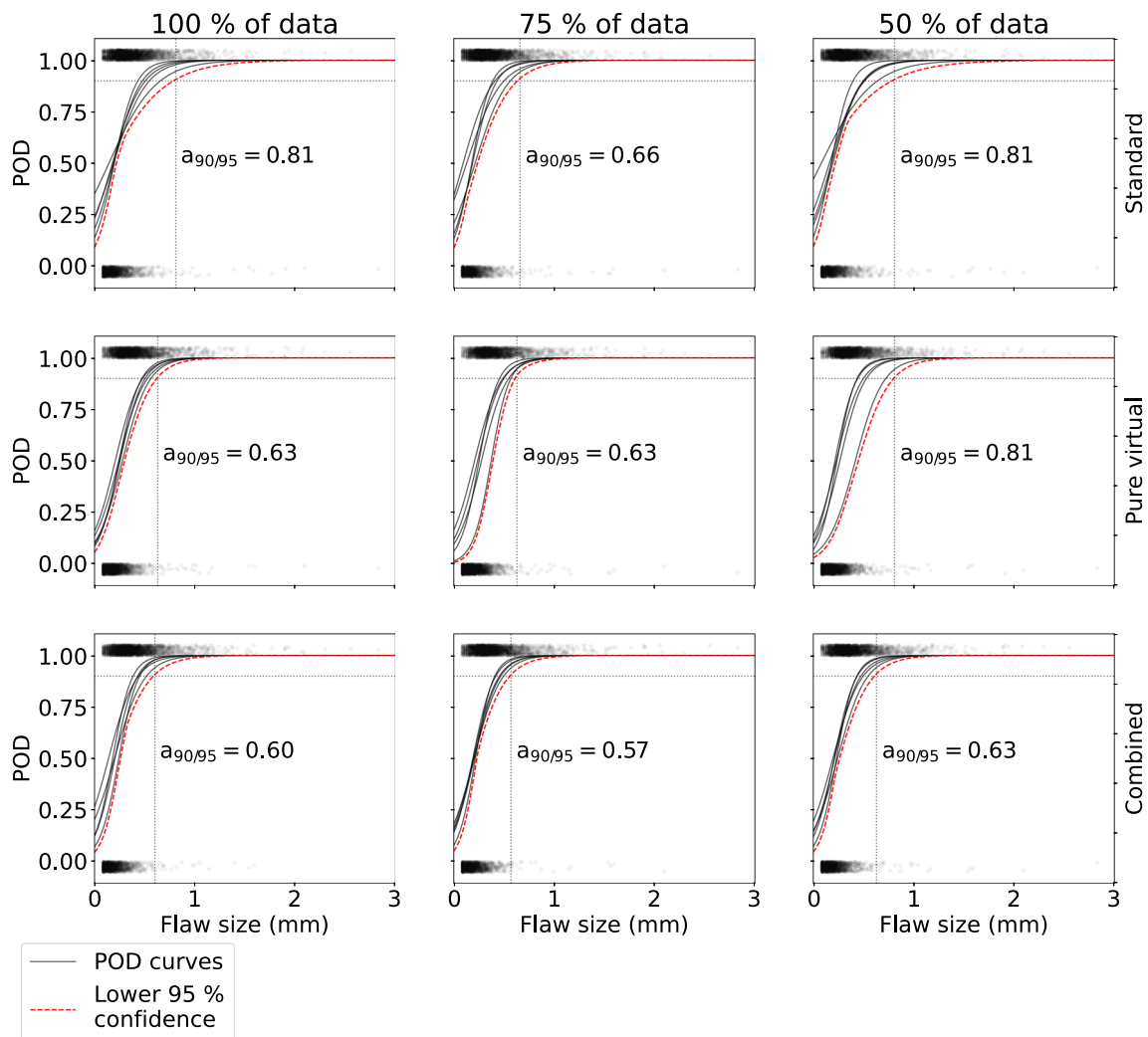

Fig. 14 POD curves for the three augmentation methods for decreasing data set size. As a result of 5-fold cross-validation, 5 separate POD curves are drawn in each figure, each indicating a separately trained and validated model. The number of defects used for testing each model is shown in Table 2. Hits are marked by black dots on the top of the plot, and misses on the bottom. The dashed curve is calculated by taking the minimum of the lower 95% confidence bounds of each POD curve. The intersection of the lower confidence bound with POD = 0.9, marked by the dotted horizontal and vertical lines, is the worst case $a_{90/95}$

Fig. 15 POD curves for the three augmentation methods for decreasing data set size. As a result of 5-fold cross-validation, 5 separate POD curves are drawn in each figure, each indicating a separately trained and validated model. The number of defects used for testing each model is shown in Table 3. Hits are marked by black dots on the top of the plot, and misses on the bottom. The dashed curve is calculated by taking the minimum of the lower 95% confidence bounds of each POD curve. The intersection of the lower confidence bound with POD = 0.9, marked by the dotted horizontal and vertical lines, is the worst case $a_{90/95}$
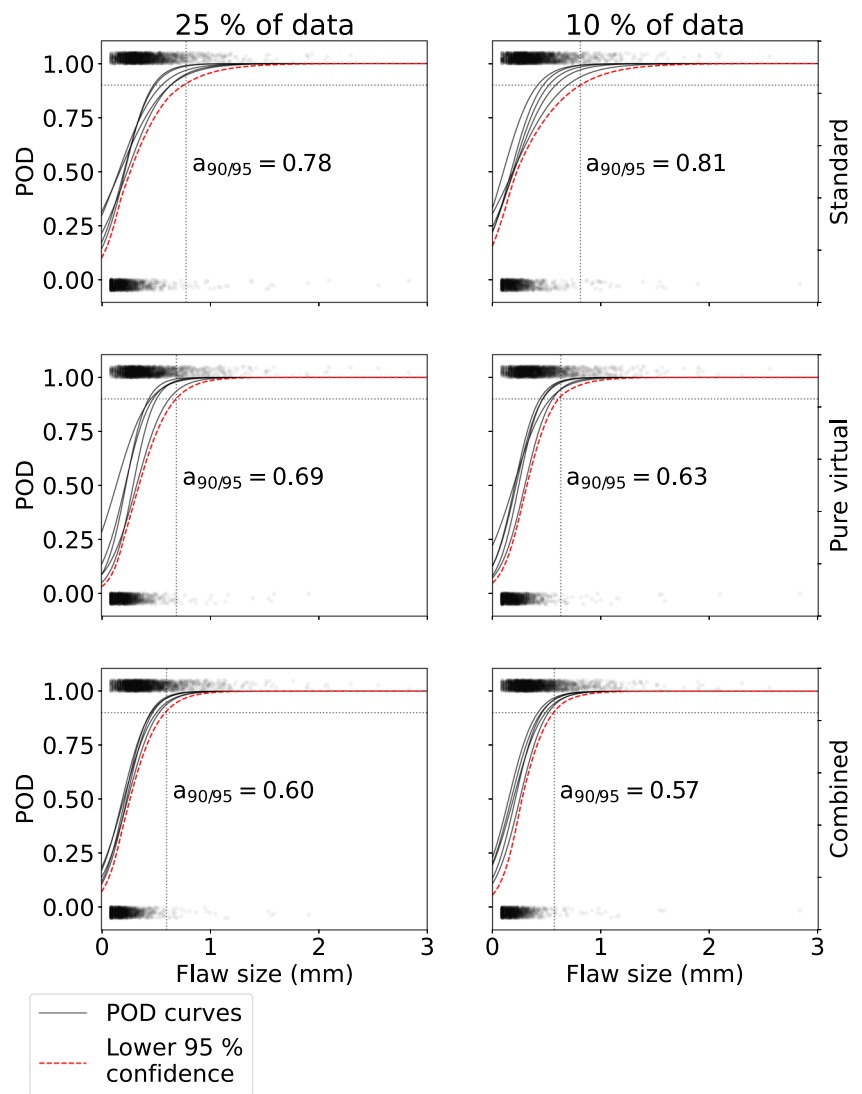
Fig. 16 POD curves for the three augmentation methods for decreasing data set size. As a result of 5-fold cross-validation, 5 separate POD curves are drawn in each figure, each indicating a separately trained and validated model. The number of defects used for testing each model is shown in Table 4. Hits are marked by black dots on the top of the plot, and misses on the bottom. The dashed curve is calculated by taking the minimum of the lower 95% confidence bounds of each POD curve. The intersection of the lower confidence bound with POD = 0.9, marked by the dotted horizontal and vertical lines, is the worst case $a_{90/95}$
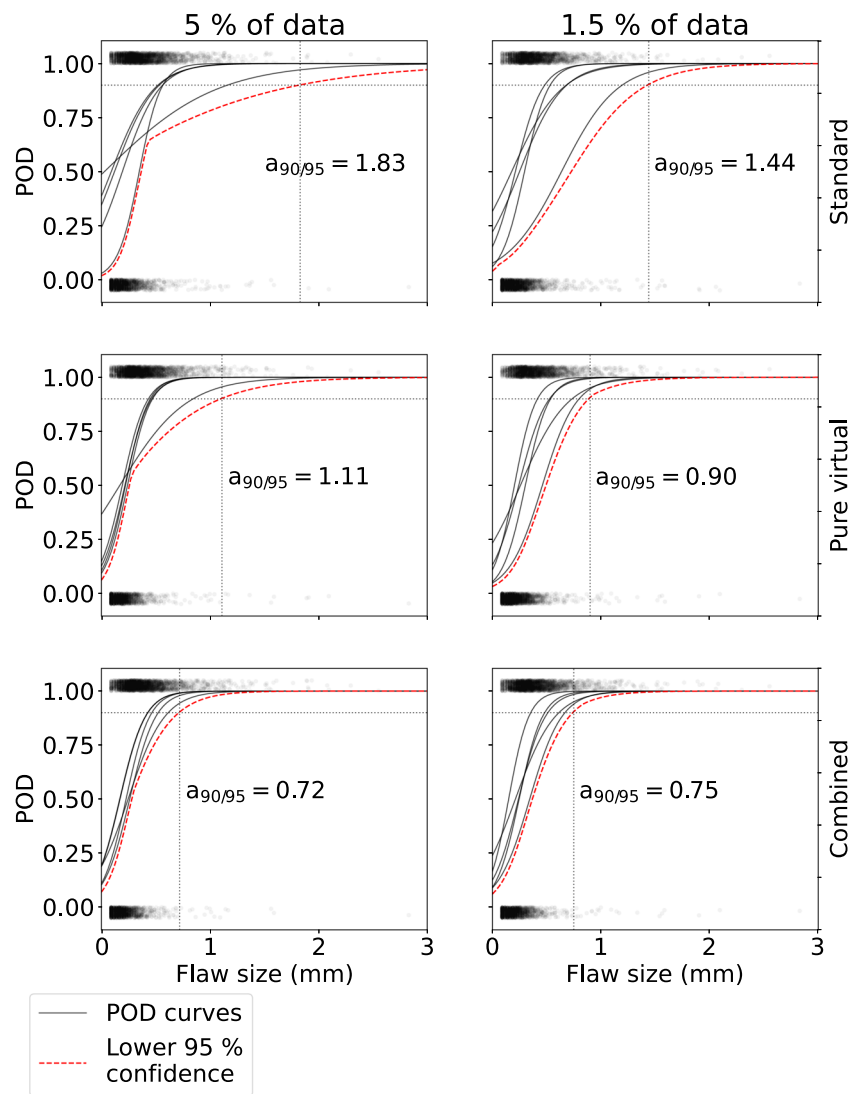
Appendix B. Full cross-validation results

The results for the cross-validation are presented in Tables 2, 3 and 4. Three augmentation methods were compared: standard, pure virtual and combined augmentation. Four metrics were recorded for each model trained: $a_{90/95}$, sizing error, false call rate on weld areas and false call rate on entire images. The metrics are defined in Section 2.6. Table 2 shows results for training data sets using 100%, 75% and 50% of available data. Table 3 displays results for 25% and 10% of data, and Table 4 for 5% and 1.5% of data.

The raw images had varying amounts of defects, thus the number of defects available for training varies between folds. For large data sets, this effects is small (1711 vs. 1825) as shown in the second column of Table 2. Table 4, however, shows significant relative differences in the number of defects at small data set sizes. Even at small data sets, the variance between folds is not directly related to the defect counts: For instance at 1.5% of data, standard augmentation in fold 1 with 8 defects achieved a better $a_{90/95}$ (0.60 mm) than in fold 4 with 26 defects (0.90 mm).

Table 2 Cross-validation results for $a_{90/95}$, sizing error, false call rate (denoted FCR) on weld and FCR on image for three augmentation methods: standard, pure virtual and combined augmentation. Results for 100%, 75% and 50% of available training data used

	Train set defects	$a_{90/95}$ (mm)	Sizing error (mm)	FCR, weld (1/cm)	FCR, image (1/cm)
100% of training data					
Standard 1	1815	0.63	0.10	0.05	0.22
Standard 2	1825	0.81	0.11	0.08	0.29
Standard 3	1753	0.48	0.11	0.05	0.28
Standard 4	1775	0.54	0.09	0.04	0.38
Standard 5	1711	0.57	0.10	0.04	0.13
Pure virtual 1	1815	0.60	0.09	0.05	0.28
Pure virtual 2	1825	0.63	0.12	0.03	0.62
Pure virtual 3	1753	0.57	0.10	0.05	0.39
Pure virtual 4	1775	0.54	0.13	0.09	0.58
Pure virtual 5	1711	0.54	0.10	0.05	0.26
Combined 1	1815	0.60	0.10	0.06	0.23
Combined 2	1825	0.42	0.13	0.12	0.46
Combined 3	1753	0.51	0.09	0.03	0.29
Combined 4	1775	0.48	0.11	0.08	0.36
Combined 5	1711	0.51	0.11	0.10	0.30
75% of training data					
Standard 1	1276	0.48	0.11	0.06	0.28
Standard 2	1364	0.48	0.11	0.16	0.44
Standard 3	1284	0.66	0.09	0.04	0.18
Standard 4	1297	0.45	0.11	0.09	0.39
Standard 5	1331	0.57	0.12	0.10	0.59
Pure virtual 1	1276	0.51	0.10	0.04	0.43
Pure virtual 2	1364	0.57	0.10	0.10	0.28
Pure virtual 3	1284	0.60	0.17	0.08	1.00
Pure virtual 4	1297	0.63	0.13	0.02	0.26
Pure virtual 5	1331	0.57	0.12	0.06	0.84
Combined 1	1276	0.48	0.10	0.12	0.59
Combined 2	1364	0.57	0.10	0.05	0.17
Combined 3	1284	0.54	0.10	0.05	0.22
Combined 4	1297	0.45	0.10	0.05	0.29
Combined 5	1331	0.51	0.12	0.07	0.38
50% of training data					
Standard 1	866	0.81	0.11	0.08	0.25
Standard 2	783	0.57	0.11	0.04	0.21
Standard 3	921	0.57	0.08	0.03	0.25
Standard 4	864	0.60	0.15	0.07	0.39
Standard 5	950	0.45	0.13	0.07	0.40
Pure virtual 1	866	0.81	0.16	0.02	0.83
Pure virtual 2	783	0.51	0.12	0.06	0.34
Pure virtual 3	921	0.60	0.12	0.03	0.94
Pure virtual 4	864	0.57	0.10	0.04	0.29
Pure virtual 5	950	0.48	0.10	0.04	0.31
Combined 1	866	0.63	0.12	0.06	0.70
Combined 2	783	0.48	0.12	0.07	0.68
Combined 3	921	0.54	0.10	0.05	0.34
Combined 4	864	0.57	0.11	0.08	0.38
Combined 5	950	0.48	0.10	0.07	0.57

Table 3 Cross-validation results for $a_{90/95}$, sizing error, false call rate (denoted FCR) on weld and FCR on image for three augmentation methods: standard, pure virtual and combined augmentation. Results for 25% and 10% of available training data used

|	Train set defects	$a_{90/95}$ (mm)	Sizing error (mm)	FCR, weld (1/cm)	FCR, image (1/cm)
25% of training data					
Standard 1	446	0.57	0.11	0.06	0.43
Standard 2	473	0.54	0.10	0.05	0.21
Standard 3	512	0.75	0.11	0.03	0.40
Standard 4	488	0.78	0.10	0.06	0.30
Standard 5	347	0.66	0.10	0.05	0.18
Pure virtual 1	446	0.51	0.11	0.17	0.66
Pure virtual 2	473	0.48	0.10	0.05	0.31
Pure virtual 3	512	0.54	0.10	0.08	0.51
Pure virtual 4	488	0.57	0.13	0.04	0.28
Pure virtual 5	347	0.69	0.10	0.12	0.92
Combined 1	446	0.57	0.09	0.05	0.30
Combined 2	473	0.51	0.11	0.06	0.86
Combined 3	512	0.51	0.12	0.16	0.53
Combined 4	488	0.54	0.12	0.05	0.31
Combined 5	347	0.60	0.12	0.05	0.44
10% of training data					
Standard 1	183	0.69	0.12	0.06	0.39
Standard 2	210	0.48	0.10	0.10	0.37
Standard 3	183	0.54	0.09	0.07	0.46
Standard 4	175	0.60	0.14	0.08	0.29
Standard 5	173	0.81	0.11	0.05	0.40
Pure virtual 1	183	0.51	0.12	0.11	0.76
Pure virtual 2	210	0.63	0.14	0.07	1.28
Pure virtual 3	183	0.51	0.10	0.07	0.25
Pure virtual 4	175	0.63	0.12	0.08	0.53
Pure virtual 5	173	0.48	0.11	0.05	0.46
Combined 1	183	0.57	0.13	0.07	0.39
Combined 2	210	0.51	0.10	0.07	0.44
Combined 3	183	0.51	0.10	0.08	0.33
Combined 4	175	0.57	0.11	0.07	0.56
Combined 5	173	0.48	0.11	0.11	0.45

Table 4 Cross-validation results for $a_{90/95}$, sizing error, false call rate (denoted FCR) on weld and FCR on image for three augmentation methods: standard, pure virtual and combined augmentation. Results for 5% and 1.5% of available training data used

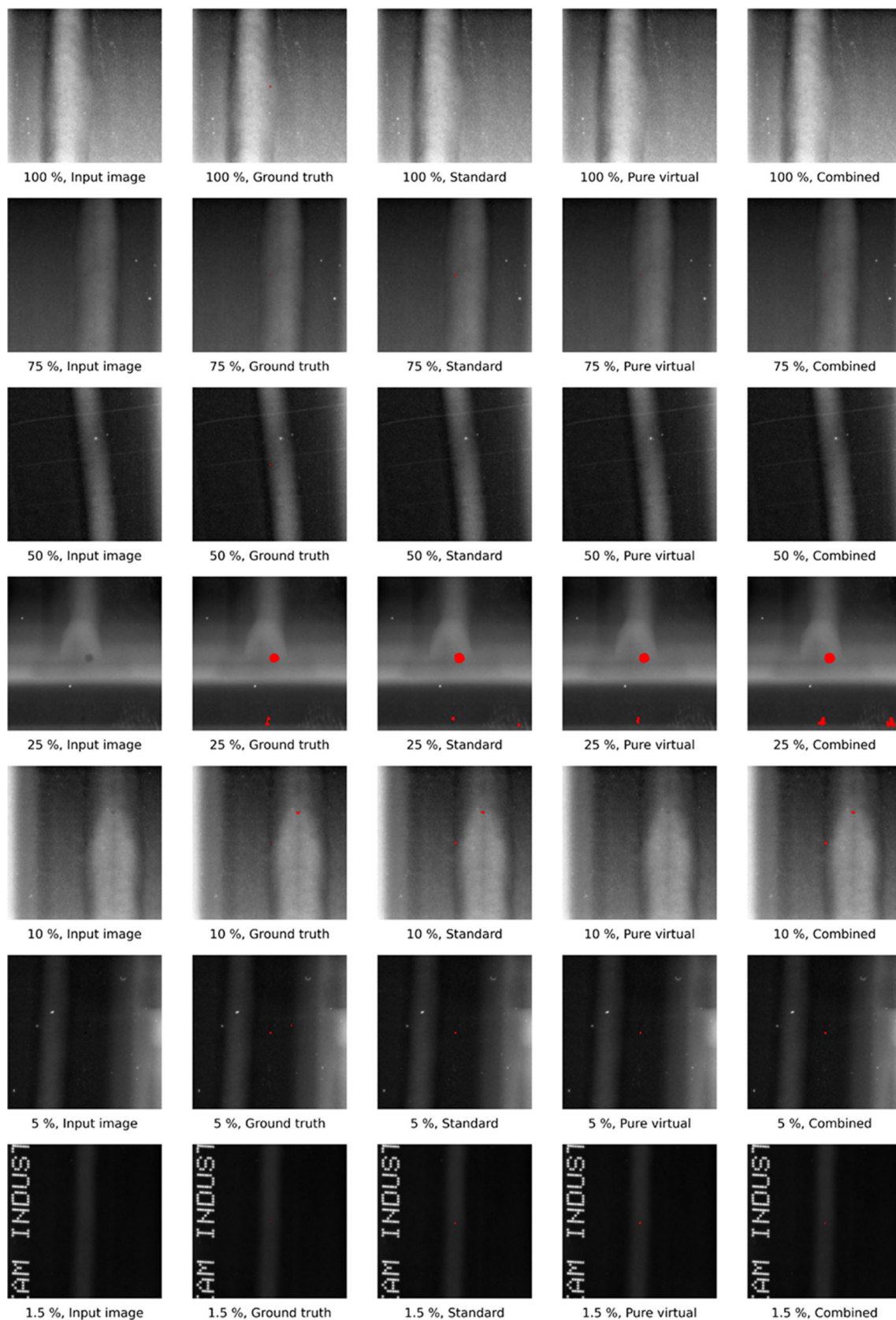|	Train set defects	$a_{90/95}$ (mm)	Sizing error (mm)	FCR, weld (1/cm)	FCR, image (1/cm)
5% of training data					
Standard 1	121	0.63	0.14	0.29	0.93
Standard 2	119	0.66	0.10	0.07	0.21
Standard 3	159	1.83	0.14	0.15	0.63
Standard 4	73	0.63	0.11	0.13	0.40
Standard 5	91	0.63	0.20	0.04	0.57
Pure virtual 1	121	0.48	0.12	0.10	0.70
Pure virtual 2	119	0.54	0.12	0.06	0.69
Pure virtual 3	159	1.11	0.13	0.07	0.45
Pure virtual 4	73	0.51	0.08	0.09	0.69
Pure virtual 5	91	0.51	0.18	0.30	1.34
Combined 1	121	0.48	0.11	0.07	0.30
Combined 2	119	0.60	0.13	0.06	0.36
Combined 3	159	0.72	0.12	0.03	0.32
Combined 4	73	0.48	0.09	0.10	0.90
Combined 5	91	0.51	0.14	0.07	0.44
1.5% of training data					
Standard 1	8	0.60	0.16	0.02	1.02
Standard 2	7	0.81	0.14	0.05	0.24
Standard 3	17	1.44	0.26	0.01	0.21
Standard 4	26	0.81	0.13	0.06	1.08
Standard 5	13	0.54	0.15	0.12	0.97
Pure virtual 1	8	0.60	0.11	0.04	1.31
Pure virtual 2	7	0.60	0.13	0.11	0.31
Pure virtual 3	17	0.87	0.17	0.23	0.45
Pure virtual 4	26	0.90	0.14	0.02	2.01
Pure virtual 5	13	0.48	0.12	0.26	2.55
Combined 1	8	0.54	0.13	0.10	2.38
Combined 2	7	0.57	0.19	0.12	0.60
Combined 3	17	0.72	0.11	0.14	0.56
Combined 4	26	0.75	0.12	0.03	0.86
Combined 5	13	0.39	0.18	0.65	3.15

Appendix C. Prediction masks

A random sampling of image patches, ground truths and predicted segmentation masks are shown in Figs. 17 and 18.

Each of them presents results for seven input images: one for each fraction of available data used for training, ranging from 100 to 1.5%.
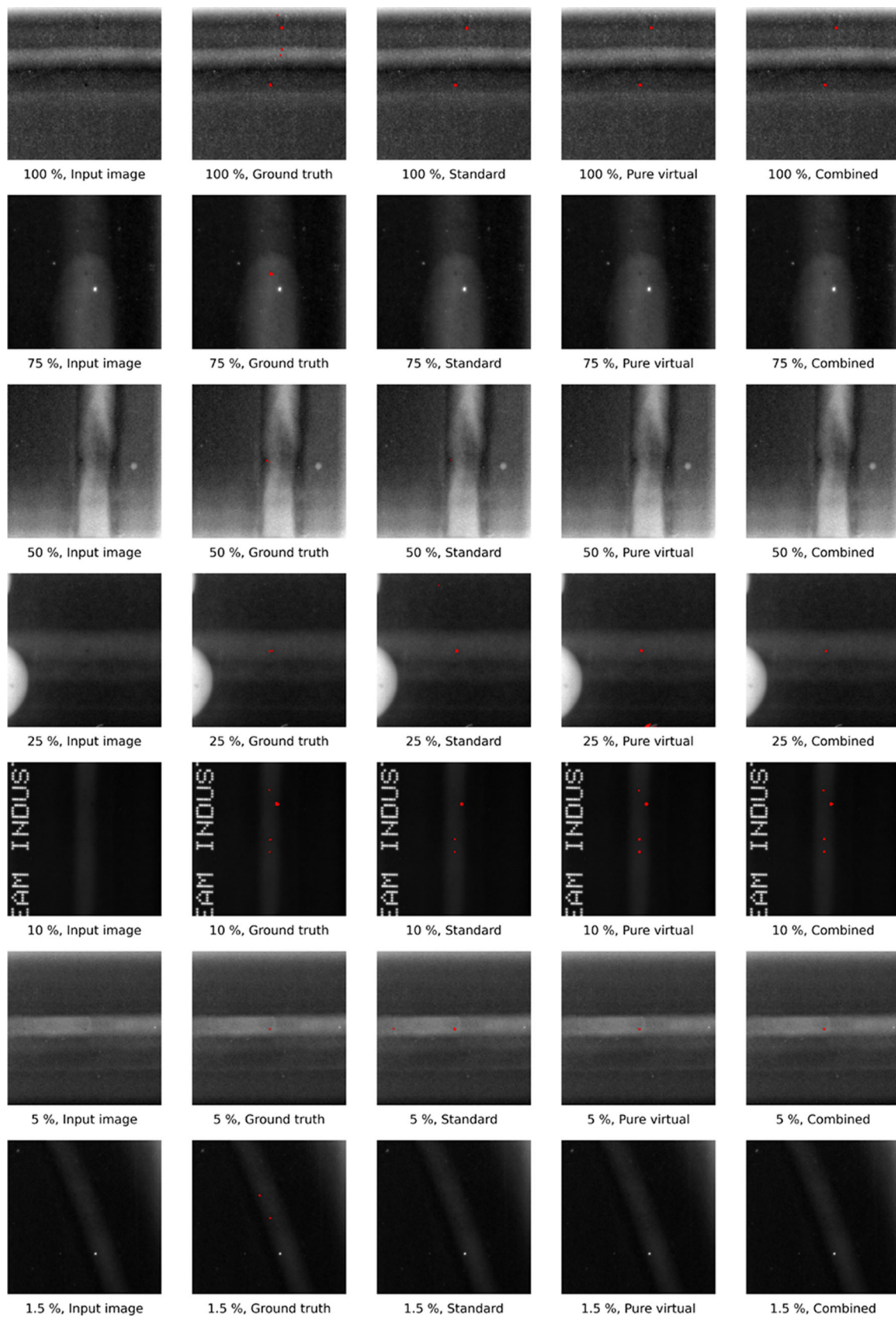
Fig. 17 Randomly extracted patches from inference results of the different augmentation methods for training data set sizes ranging from 100 to 1.5%. Input images and ground truth masks are shown on the first and second columns, respectively. The predicted masks from the standard, pure virtual and combined augmentations are shown in columns 3, 4 and 5

Fig. 18 Randomly extracted patches from inference results of the different augmentation methods for training data set sizes ranging from 100 to 1.5%. Input images and ground truth masks are shown on the first and second columns, respectively. The predicted masks from the standard, pure virtual and combined augmentations are shown in columns 3, 4 and 5

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Annis C (2009) Mil-hdbk-1823a, nondestructive evaluation system reliability assessment

2. ASTM International (2021) Standard practice for probability of detection analysis for â versus a data (astm e3023-21). https://doi.org/10.1520/E3023-21

3. Badue C, Guidolini R, Carneiro RV, Azevedo P, Cardoso VB, Forechi A, Jesus L, Berriel R, Paixao TM, Mutz F et al (2020) Self-driving cars: a survey. Expert Syst Appl: 113816. https://doi.org/10.1016/j.eswa.2020.113816

4. Bertović M (2016) Human factors in non-destructive testing (ndt): risks and challenges of mechanised ndt. PhD thesis, Technische Universitaet Berlin (Germany), https://doi.org/10.14279/depositonce-4685

5. Bertovic M, Virkkunen I (2021) NDE 4.0: new paradigm for the NDE inspection personnel, pp 1–31. Springer International Publishing. https://doi.org/10.1007/978-3-030-48200-8

6. Bochkovskiy A, Wang CY, Liao HYM (2020) Yolov4: optimal speed and accuracy of object detection. arXiv:200410934

7. Canny J (1986) A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence PAMI 8(6):679–698. https://doi.org/10.1109/TPAMI.1986.4767851

8. Chapuis B, Jenson F, Calmon P, DiCrisci G, Hamilton J, Pomié L (2014) Simulation supported pod curves for automated ultrasonic testing of pipeline girth welds. Welding in the World 58(4):433–441. https://doi.org/10.1007/s40194-014-0125-z

9. Chapuis B, Calmon P, Jenson F et al (2016) Best practices for the use of simulation in pod curves estimation. IIW Collection https://doi.org/10.1007/978-3-319-62659-8

10. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. IEEE, pp 248–255. https://doi.org/10.1109/CVPR.2009.5206848

11. Du W, Shen H, Fu J, Zhang G, He Q (2019) Approaches for improvement of the x-ray image defect detection of automobile casting aluminum parts based on deep learning. NDT & E International 107:102,144. https://doi.org/10.1016/j.ndteint.2019.102144

12. Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A (2011) The PASCAL visual object classes challenge 2011 (VOC2011) results. http://www.pascal-network.org/challenges/VOC/voc2011/workshop/index.html

13. Gamdha D, Unnikrishnakurup S, Rose KJ, Surekha M, Purushothaman P, Ghose B, Balasubramaniam K (2021) Automated defect recognition on x-ray radiographs of solid propellant using deep learning based on convolutional neural networks. J Nondestruct Eval 40(1):1–13. https://doi.org/10.1007/s10921-021-00750-4

14. Greene A, Michael M, JJM III, Betz R, Barry R, Nightingale G, Siewert TA, Anderson CE, Luga TF, Folland WH, Surma G, McCullough R, Thams RW, Apgar B, Becker G, McKinney WE, Wenk SA, 1992 ASM handbook. Volume 17, Nondestructive evaluation and quality control. Radiographic inspection. ASM International. https://doi.org/10.31399/asm.hb.v17.9781627081900

15. Guo Z, Hall RW (1992) Fast fully parallel thinning algorithms. CVGIP: Image Understanding 55(3):317–328. https://doi.org/10.1016/1049-9660(92)90029-3

16. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778. https://doi.org/10.1109/CVPR.2016.90

17. He K, Gkioxari G, Dollár P, Girshick R (2017) Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision, pp 2961–2969. https://doi.org/10.1109/ICCV.2017.322

18. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708. https://doi.org/10.1109/CVPR.2017.243

19. International Organization for Standardization (2007) Welding and allied processes – classification of geometric imperfections in metallic materials – part 1: fusion welding (iso 6520-1:2007)

20. International Organization for Standardization (2014) Welding – fusion-welded joints in steel, nickel, titanium and their alloys (beam welding excluded) – quality levels for imperfections (iso 5817:2014)

21. International Organization for Standardization (2016) Non-destructive testing of welds – acceptance levels for radiographic testing – part 1: steel, nickel, titanium and their alloys (iso 10675-1:2016)

22. Irvin J, Rajpurkar P, Ko M, Yu Y, Ciurea-Ilcus S, Chute C, Marklund H, Haghgoo B, Ball R, Shpanskaya K et al (2019) Chexpert: a large chest radiograph dataset with uncertainty labels and expert comparison. In: Proceedings of the AAAI conference on artificial intelligence, vol 33, pp 590–597. https://doi.org/10.1609/aaai.v33i01.3301590

23. Jiang H, Hu Q, Zhi Z, Gao J, Gao Z, Wang R, He S, Li H (2021) Convolution neural network model with improved pooling strategy and feature selection for weld defect recognition. Welding in the World 65(4):731–744. https://doi.org/10.1007/s40194-020-01027-6

24. Jonsson B, Dobmann G, Hobbacher A, Kassner M, Marquis G (2016) IIW guidelines on weld quality in relationship to fatigue strength. Springer. https://doi.org/10.1007/978-3-319-19198-0

25. Kimpe T, Tuytschaever T (2007) Increasing the number of gray shades in medical display systems—how much is enough? J Digit Imaging 20(4):422–432. https://doi.org/10.1007/s10278-006-1052-3

26. Konnik M, Ahmadi B, May N, Favata J, Shahbazi Z, Shahbazmohamadi S, Tavousi P (2021) Training ai-based feature extraction algorithms, for micro ct images, using synthesized data. J Nondestruct Eval 40(1):1–13. https://doi.org/10.1007/s10921-021-00758-w

27. Koskinen T, Virkkunen I, Siljama O, Jessen-Juhler O (2021) The effect of different flaw data to machine learning powered ultrasonic inspection. J Nondestruct Eval 40(1):1–13. https://doi.org/10.1007/s10921-021-00757-x

28. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 25:1097–1105. https://doi.org/10.1145/3065386

29. Li Z, Wang C, Han M, Xue Y, Wei W, Li LJ, Fei-Fei L (2018) Thoracic disease identification and localization with limited supervision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8290–8299. https://doi.org/10.1109/CVPR.2018.00865

30. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2117–2125. https://doi.org/10.1109/CVPR.2017.106

31. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3431–3440. https://doi.org/10.1109/CVPR.2015.7298965

32. Mery D (2011) Automated detection of welding discontinuities without segmentation. Mater Eval 69(6):656–663

33. Mery D (2021) Aluminum casting inspection using deep object detection methods and simulated ellipsoidal defects. Mach Vis Appl 32(3):1–16. https://doi.org/10.1007/s00138-021-01195-5

34. Mery D, Arteta C (2017) Automatic defect recognition in x-ray testing using computer vision. In: 2017 IEEE winter conference on applications of computer vision (WACV). IEEE, pp 1026–1035. https://doi.org/10.1109/WACV.2017.119

35. Mery D, Riffo V, Zscherpel U, Mondragón G, Lillo I, Zuccar I, Lobel H, Carrasco M (2015) Gdxray: the database of x-ray images for nondestructive testing. J Nondestruct Eval 34(4):1–12. https://doi.org/10.1007/s10921-015-0315-7

36. Nacereddine N, Zelmat M, Belaifa SS, Tridi M (2005) Weld defect detection in industrial radiography based digital image processing. Trans Eng Comput Technol 2:145–148. https://doi.org/10.5281/zenodo.1330641

37. NASA (2019) Nasa-std-5009b, nondestructive evaluation requirements for fracture-critical metallic components

38. Rajpurkar P, Irvin J, Zhu K, Yang B, Mehta H, Duan T, Ding D, Bagul A, Langlotz C, Shpanskaya K et al (2017) Chexnet: radiologist-level pneumonia detection on chest x-rays with deep learning. arXiv:171105225

39. Redmon J, Farhadi A (2018) Yolov3: an incremental improvement. arXiv:180402767

40. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 779–788. https://doi.org/10.1109/CVPR.2016.91

41. Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: towards real-time object detection with region proposal networks. arXiv:150601497. https://doi.org/10.1109/TPAMI.2016.2577031

42. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

43. Rowlands J (2002) The physics of computed radiography. Physics in Medicine & Biology 47(23):R123. https://doi.org/10.1088/0031-9155/47/23/201

44. Saez D (2004) Automated defect detection in aluminium castings and welds using neuro-fuzzy classifiers. In: 16th World conference on nondestructive testing. Citeseer

45. Schwartz C (2003) Automatic evaluation of welded joints using image processing on radiographs. In: AIP Conference proceedings, vol 657. American Institute of Physics, pp 689–694. https://doi.org/10.1063/1.1570203

46. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv:14091556

47. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9. https://doi.org/10.1109/CVPR.2015.7298594

48. Virkkunen I (2021) The "small crack problem" in hit/miss probability of detection. Unpublished

49. Virkkunen I, Miettinen K, Packalen T (2014) Virtual flaws for nde training and qualification. In: 11th European conference on non-destructive testing (ECNDT 2014)

50. Virkkunen I, Koskinen T, Jessen-Juhler O, Rinta-Aho J (2021) Augmented ultrasonic data for machine learning. J Nondestruct Eval 40(1):1–11. https://doi.org/10.1007/s10921-020-00739-5

51. Wang X, Peng Y, Lu L, Lu Z, Bagheri M, Summers RM (2017) Chestx-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2097–2106. https://doi.org/10.1109/CVPR.2017.369

52. Xu N, Price B, Cohen S, Huang T (2017) Deep image matting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2970–2979. https://doi.org/10.1109/CVPR.2017.41

53. Zhao ZQ, Zheng P, St Xu, Wu X (2019) Object detection with deep learning: a review. IEEE Trans Neural Netw Learn Syst 30(11):3212–3232. https://doi.org/10.1109/TNNLS.2018.2876865

54. Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A (2016) Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2921–2929. https://doi.org/10.1109/CVPR.2016.319

Affiliations

Topias Tyystjärvi[1,2] · Iikka Virkkunen[1] · Peter Fridolf[3] · Anders Rosell[3] · Zuheir Barsoum[4]

Iikka Virkkunen
iikka.virkkunen@trueflaw.com

Peter Fridolf
peter.fridolf@gknaerospace.com

Anders Rosell
anders.ar.rosell@gknaerospace.com

Zuheir Barsoum
zuheir@kth.se

1 Trueflaw, Tillinmäentie 3 A 113, FI-02330 Espoo, Finland

2 Department of Mechanical Engineering, School of Engineering, Aalto University, P.O. Box 14200, FI-00076 Aalto, Finland

3 GKN Aerospace Engine Systems, SE-46138 Trollhättan, Sweden

4 Department of Engineering Mechanics, KTH Royal Institute of Technology, Teknikringen 8, SE-100 44 Stockholm, Sweden