Automated Defect Recognition on X-ray Radiographs of Solid Propellant Using Deep Learning Based on Convolutional Neural Networks

Dhruv Gamdha[1] · **Sreedhar Unnikrishnakurup**[1] · **K. J. Jyothir Rose**[2] · **M. Surekha**[2] · **Padma Purushothaman**[2] · **Bikash Ghose**[3] · **Krishnan Balasubramaniam**[1]

Abstract

For defense applications, rapid X-ray inspection of propellant samples is essential for the identification and assessment of defects. Automation of this process using artificial intelligence is possible by properly training a neural network model. Convolution Neural Networks (CNNs) have recently demonstrated excellent success in both the tasks of image recognition and localisation using an adequate amount of data. In real-world, it's not an easy task to produce the correct amount of experimental data required for the deep neural network to operate. In this work, we propose a method for producing synthetic radiographic data that is supported by ray tracing based radiographic simulations for the deep learning algorithms to automatically detect anomaly in X-ray images. The simulation results, which are then supplemented by noise extracted from the experimental data, show a good comparison with the measurements. This Simulation assisted Automatic Defect Recognition (Sim-ADR) system simultaneously perform defect detection and defect instance segmentation. The accuracy of the defect detection system is more than 87% on a testing set included 416 images.

1 Introduction

Solid propellant is a type of high-energy material widely used in the aerospace and military sectors. The solid propellant is formulated in a special structure/shape to meet the necessary ballistic output and is known as propellant grains [1,2]. Owing to the inherent complexities, the propellant grain produced goes through rigorous quality assurance processes. To examine the effects and reasons for the defects, identification and characterization of the defects present in propellant grain in terms of their position and size is important. X-ray radiography [3] is the commonly used non-destructive measurement tool for assuring propellant grain quality [4]. Till recent past, the conventional film radiography was dominating the process in which the components used to be manually placed within the X-ray field and placement of the film loaded cassette to capture the transmitted X-ray radiation. The process used to be time consuming and the quality assurance requires expert human analysis of the X-ray defect features. For long, this process used to be the bottleneck for the higher throughput of production. After advent of digital radiography systems [5–7], which involves either an off-line or an on-line process, where components are still manually placed within the X-ray field of view (FOV) but the digital image of the transmitted X-rays is obtained with very less exposure time. These digital radiographs are also visually evaluated/interpreted, the visual evaluation is not only very much tedious due to the voluminous work but also less reliable because of fatigue of human interpreter [8]. Hence the automation of the process is critical to achieve high reliability and speed of inspection.

Several researchers explored the identification of defects from radiographic images using different techniques and

✉ Dhruv Gamdha
dhruvgamdha@gmail.com

1   Center for Non-destructive Evaluation, Indian Institute of Technology Madras, Chennai, India

2   Dhvani Research, Chennai, India

3   High Energy Material Research Laboratory, Pune, India

approaches [9–14]. Artificial intelligence and computer vision methods can be used to aid in analyzing the X-rays and provide an indication of the examined material's diagnosis. There has been a lot of effort to build and construct computational tools focused on image processing, computer vision, artificial intelligence and other related fields with the aim of encouraging radiograph analysis and thus improving the robustness, accuracy and speed of the inspection process [15–23]. Owing to the exponential increase in processing algorithms and computing power, Image classification, object detection, and image segmentation fields are advancing significantly. The development of deep Convolution Neural Networks (CNN) has led to major improvements in several tasks related to image processing. Ferguson et. al. [24] demonstrated the defect detection system exceeds state-of-the-art performance for defect detection on the GDXray [25] casting dataset using transfer learning technique on MaskRCNN deep learning model. In this work, we present the accomplishment of an automated defect recognition (ADR) system that uses deep learning algorithms to improve the effectiveness of the automated data analysis. The defect detection system used in this work is based on Mask Region-based CNN [26] that simultaneously perform object detection and image segmentation.

The key technology in this work is the development of advanced Simulation assisted Automated Defect Recognition (SimADR) algorithms for flaw detection, classification and characterizations of different types of propellant grains and rocket motors. These propellants are likely to develop flaws at the interfaces and in the grains due to perturbations in the process variables and consequently its manifestation in the final products are inherent in a mass manufacturing

environment due to several reasons. Several forms of simulation software have been developed over the past few years for radiological applications [27–30]. In this work, the method is based on creating a radiograph database with different potential defect characteristics using a simulation model based on ray casting and using this to train a deep learning algorithm.

2 Materials and Methods

The Sim-ADR system is proposed to identify the defects from the propellant grain X-ray images. Figure 1 presents the Sim-ADR development flowchart. The proposed system consists of three main stages; a pre-processor, Simulation engine for X-ray images (Sim-Xray) and Artificial Neural Network based Automatic Defect Recognition (ANN-ADR). Each stage consists of a variety of processes and communicate correctly with the input of the next stage before the final identification report is obtained. In the following subsections, the functionality of each stage is described in more detail.

2.1 Pre-processor and Annotation Tool Box

The Pre-processor unit prepares the radiographic image for the annotation task by enhancing the defect features in the image. Figure 2 presents the different steps and the respective tasks associated with Pre-Processor and annotation tool box. The 16-bit X-ray image input first goes through a normalization step. Algorithms to enhance the images are implemented using a combination of many Image processing filters like Sharpening filters, Contrast Limited Adaptive Histogram Equalization (CLAHE) and FFT-Low Pass filters,
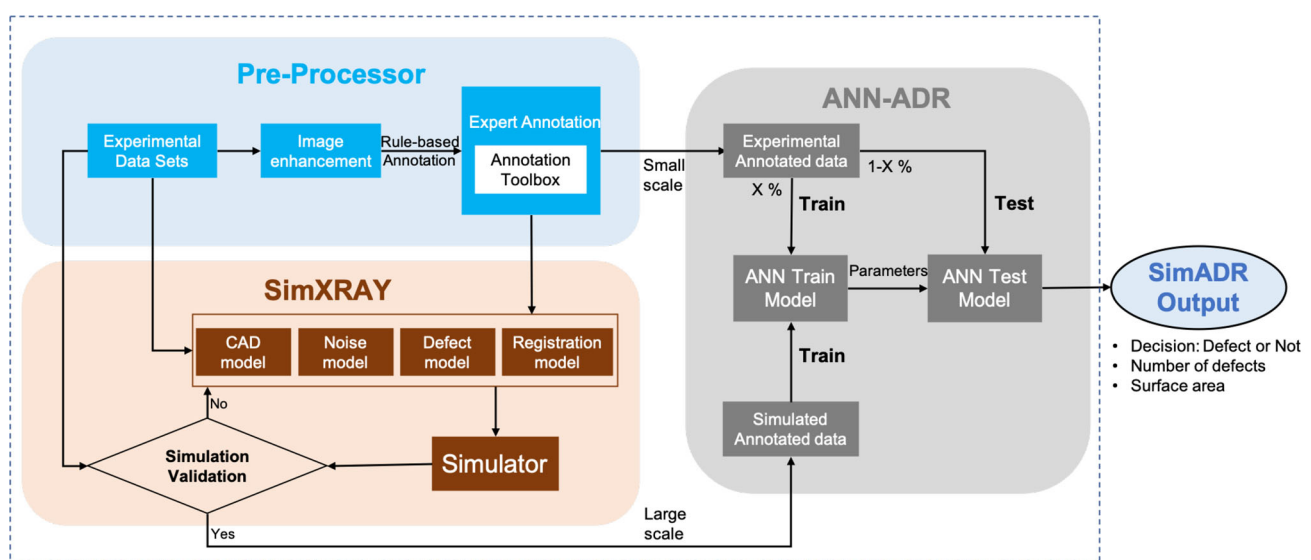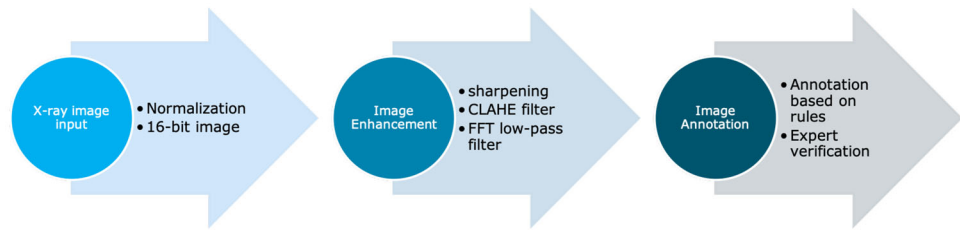

Fig. 1 SimADR system Development Flowchart

Fig. 2 Work flow chart from Pre-processor unit to annotation tool box for image enhancement and annotation process

such that the defects are clearly seen for the annotation. In the Annotation tool box, the defects are marked using the built-in annotation features like magicwand, contour and circulation and the annotated data are stored in a suitable format to train the Neural network. This is followed by an automatic segmentation of defect region and annotation using the defined rules. On the successful verification of the annotations by the Radiography Expert, the annotated images are sent to the ANN-ADR module.

2.2 Sim-XRAY

Within this section, we present the specifics of the algorithms used to generate each simulated X-ray image and ultimately generate a simulated data set for deep learning. Figure 3 displays the various sections of the Sim-Xray simulation algorithm in detail. In the object section involves the preparation of the CAD model of both component geometry and the desired artificial defects. The CAD model is then placed in between the virtual X-ray source and detector using necessary transformations. The source and detector section describes the source and detector-related parameters such as Voltage (kV), Current (mA), and exposure time. The location of the defects, the location of the source and detector, the translation and rotation associated with each measurement are included in the Registration section. The material section defines the material properties associated with the component and defects such as the attenuation coefficients and energy values. Sim-XRAY has an extensive collection of materials in its library. Further, computer graphics based Ray Casting Technique is used to generate the simulated X-Ray images using the information like Source-to-Detector distance, Source-to-Object distance, Voltage and Current value used in the measurements. Further, computer graphics based ray casting technique is used to perform the calculation of the distance traveled by the X-ray inside the component and defects separately. Using the distance values, attenuation coefficient, and energy values we calculate the X-ray intensity falling on the detector to finally generate the simulated X-ray image. The following subsections discuss each aspect of the procedure.
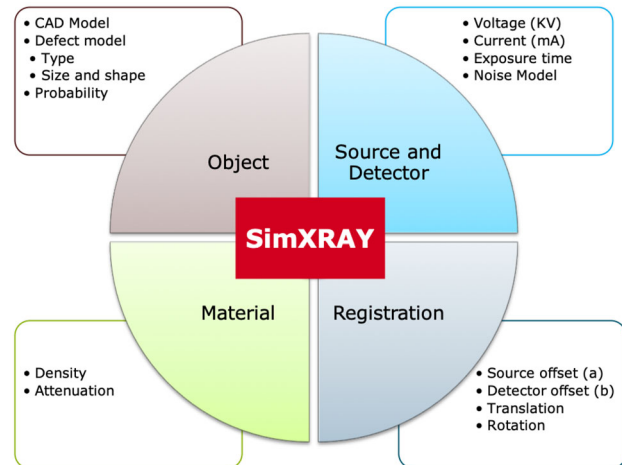

Fig. 3 SimXRAY: Simulating X-ray images

2.2.1 Preparation of the Component

The user has to prepare the CAD model of the component using any CAD software of which the X-ray is to be produced. Figure 4 shows the CAD model used in SimXRAY software. To mimic the measured X-Ray image, the sample is positioned and oriented in a way identical to the one in the experiment.

2.2.2 Addition of Artificial Defects

In this paper we focus on void defects in the casting of propellant component. The library of CAD model of defects is generated by closely observing the experimental X-Ray images obtained from the industry. These defects are caused by trapped gasses inside the composite slurry and appear as a single large bubble or a set of smaller bubbles. These defects are modelled as ellipsoid and spheroids with properties of air. The simulated radiographs are generated for the object placed at different angles. Hence flaws in 3D even if ellipsoid and spheroids will generate correct shapes on the simulated radiographs based on their location and orientations. The ellipsoids and spheroids are considered mostly because the real volumetric flaws closely replicate to ellipsoids and spheroids due to inherent casting process of solid propellant. A normal distribution is used to vary the major and minor axis of the ellipsoid or the radius in case of the
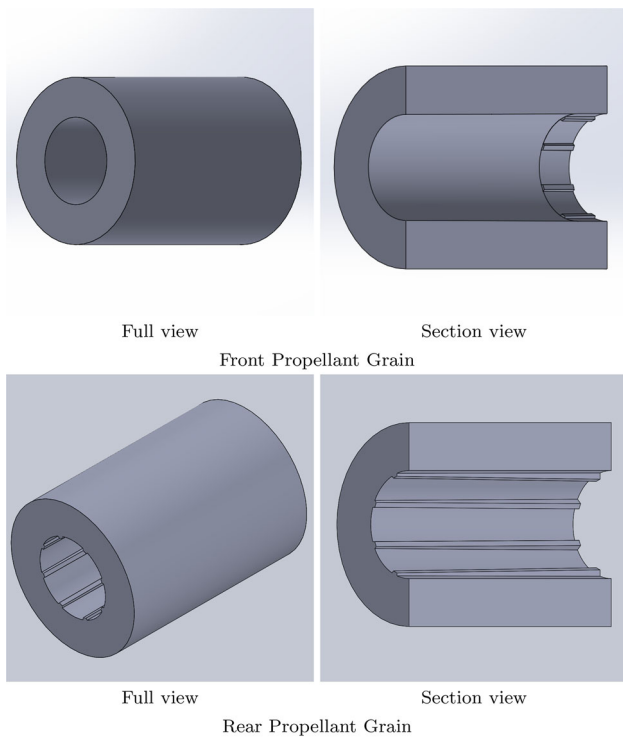
Fig. 4 CAD geometry of the propellant grain component.

sphere based on observation of the scale of the defects in comparison to the size of the component. In the case of representing the porosity as a series of bubbles, the number of bubbles is randomly selected in a range of 1 to 8 bubbles in addition to varying the radius of the spheres. Further, the position of the bubbles are chosen based on a normal distribution which penalised the bubbles being too close or too far from each other.

The defects produced are then incorporated into the CAD model of the component. The locations of the defect are bounded with in the component geometry. Figure 5 shows the flow chart to explain the defect insertion process. The component in this study is observed to have defects in the range

of 0 to 15 and follows the normal distribution. The position and orientation of the defect with respect to the component is randomly determined. The sampled position of the defect is then checked to lie completely inside the component. If not, then this process is repeated until the correct position is sampled. The defect is then translated to it's appropriate position. Figure 6 shows components with defects placed between the virtual x-ray source and detector. The defects along with their positions are stored in a list to be used during the ray casting process.

2.2.3 Ray Casting

The work by Tucker et al. [31,32] is used to perform the calculation of x-ray spectra. Using their algorithm the target x-ray intensity range is determined as a function of the energy level. The target intensity is further attenuated depending on the distance travelled by the ray inside the component. In this section we discuss the method of calculating the distance travelled by the ray inside component using the computer graphics based ray cast technique. In 3D space a point i.e. X-ray source and a plane i.e X-ray detector is defined as shown in Figs. 6 and 7. The distances between the component, detector and source is set by the user. To match the measured X-ray image the distances should be the same as that of the experimental setup. A simulation X-ray image of size $n \times m$ pixels requires, the detector plane to be divided into the same ($n \times m$) number of square boxes. Where each box on the detector corresponds to the pixel in the image. The centre of these boxes on the detector acts as the targets of the rays cast from the virtual x-ray source. Hence, for each box in the detector a ray is cast from the source and the intersection points of the rays with the component and the defects are calculated. The intersection points are further used to calculate the distance travelled by the rays inside the component and defects.

Point of intersections between the rays and the component, rays and defects are determined by separate consideration as

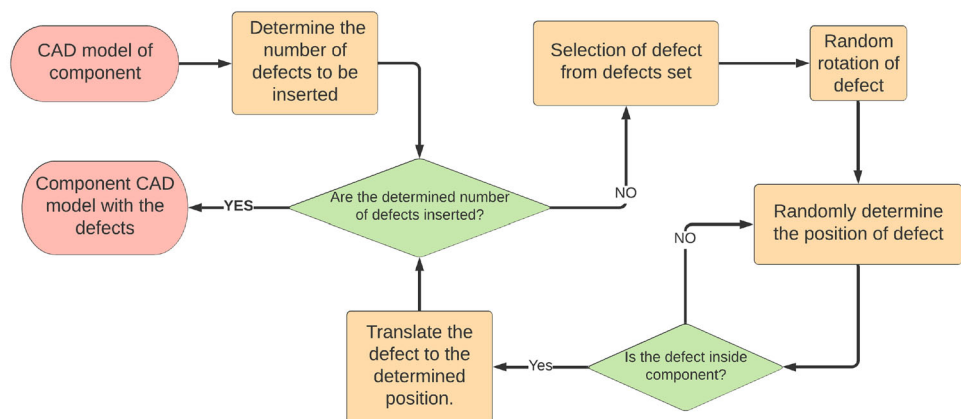Fig. 5 The defect insertion process flowchart

Fig. 6 Propellant grains with defects are placed in between the virtual flat panel X-ray detector (on left of component) and x-ray source (on right of component). The defects are segmented using red color in the image and few of them are pointed using an arrows
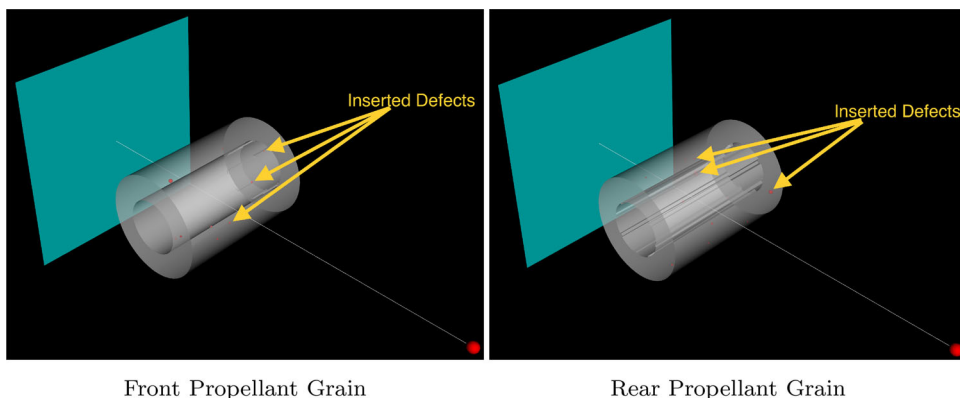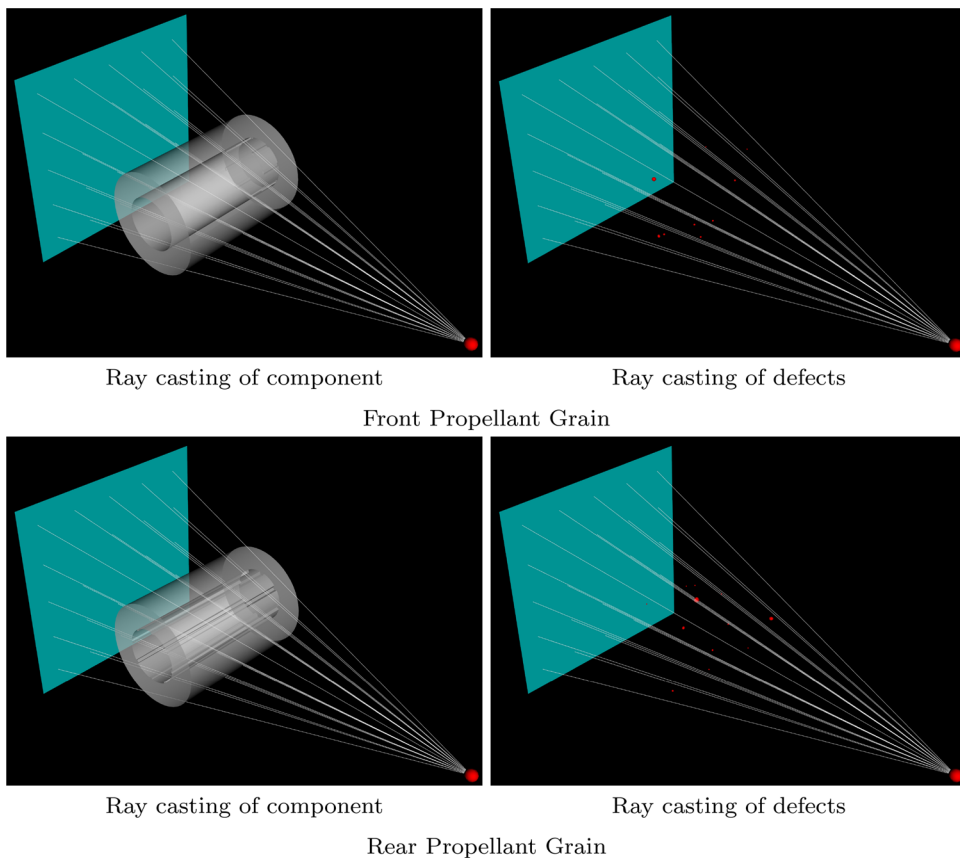
Front Propellant Grain   Rear Propellant Grain

Fig. 7 Separate ray casting of the propellant grains and its defects

Ray casting of component   Ray casting of defects
Front Propellant Grain

Ray casting of component   Ray casting of defects
Rear Propellant Grain

shown in Fig. 7. First, the intersections with the component is calculated by positioning it between the virtual source and the detector. Then the component is replaced with defects according to their predetermined locations and its intersections are calculated. For calculating these intersections the Oriented Bounding Box Tree representation of the component and defect CAD models is used.

Special cases such as the rays tangential to component or defects are handled by approximating its corresponding pixel value with the average value of surrounding pixels. For the remaining targets the intersection points are used to calculate the length of the ray travelled inside the component and defects using the following formula,

$$
D_{component/defect} = \sum_{0}^{N} distance(P_{2i}, P_{2i+1}), i = 0, 1, 2, \ldots, N/2 \quad (1)
$$

where N is the number of intersections of the ray with the CAD model of component or defects and P is the ordered list of intersection points.

The effective length travelled by the ray inside the component will be the component length minus the defect length for

Fig. 8 Two consecutive steps in the kernel movement through the image for noise mask extraction from the experimental x-ray image
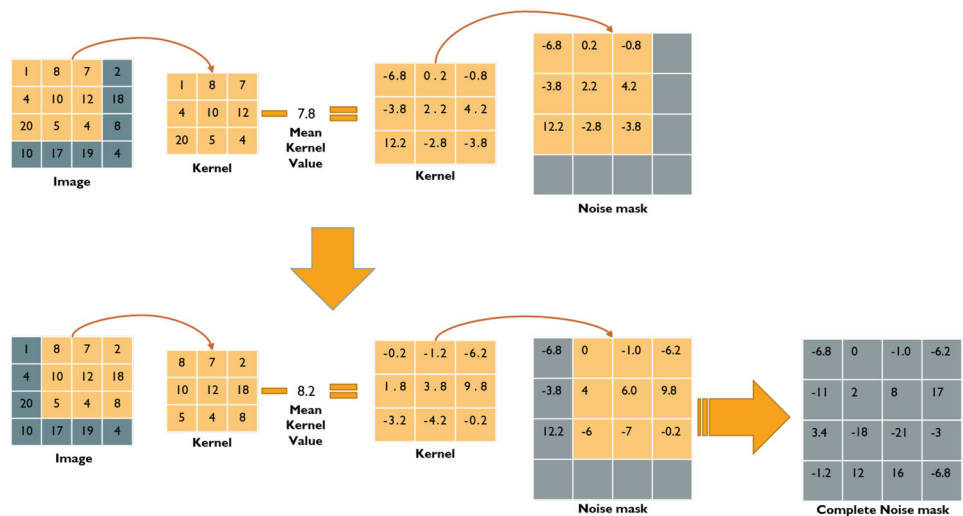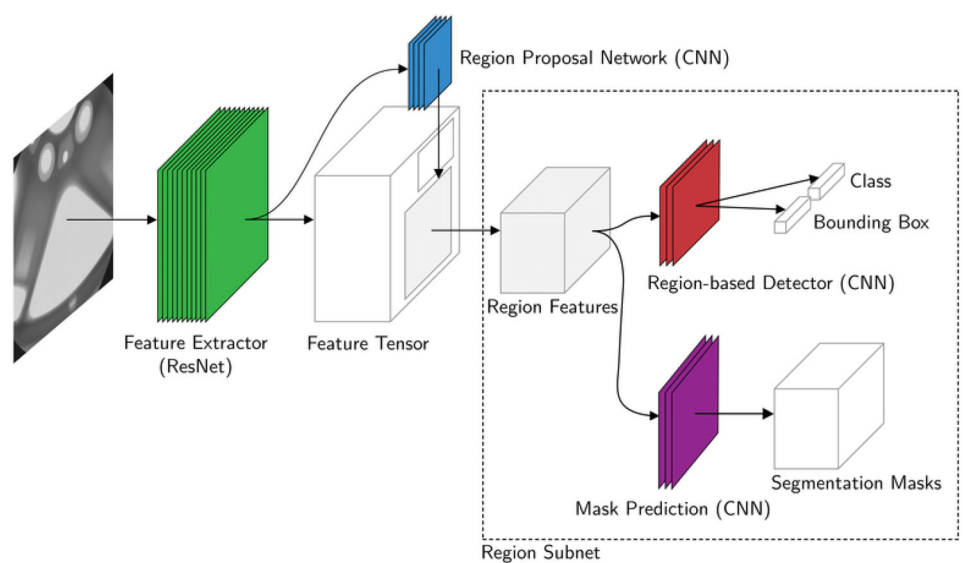

Fig. 9 The Mask R-CNN framework. The system consists of four convolutional neural networks, namely the ResNet-51 feature extractor, region proposal network (RPN), region-based detector and the mask prediction network. Reprinted from [24]

each detector target. The distance travelled by the ray inside the component and defects are stored in their respective 2D arrays each of $n \times m$ size. The two distance arrays are further used to calculate the image pixel values through a series of intensity calculations.

2.2.4 Calculation of Pixel Intensity

The distance traversed by the rays in the component and defect is used to calculate the final attenuated intensity falling on the detector pixel. The calculation is as follows,

$$I_p = \sum_{i=1}^{i=T_o} I_{o,i} *$$

$$e^{(-\mu_{component,\,i} * d_{component} - \mu_{defect,\,i} * d_{defect})}$$

(2)

where I_p is the net intensity value of the pixel after attenuation, $I_{0,i}$ is the unattenuated intensity at each energy level falling on the pixel, T_0 is energy value corresponding to current and voltage value set in the x-ray source, $\mu_{component,i}$ and $\mu_{defect,i}$ are the component and defect material's intensity attenuation coefficient respectively for each energy level, and $d_{component}$ and d_{defect} are the distance travelled by the ray inside the component and defect respectively.

The intensity of the unattenuated ray falling on the detector will be,

$$O_p = \sum_{i=1}^{i=T_o} I_{o,i}$$

(3)

where $I_{o,i}$ is the source intensity value. As each target in the detector grid represents a pixel, its pixel value is calculated by first taking square root of the ratio of I_p and O_p and

then further normalizing it between 0 to 65535 to obtain a 16bit image.

2.3 Noise Modeling

For noise modelling we are proposing a unique technique which involves extraction of noise mask from an experimentally measured image using a sliding kernel approach which separates the noise from the image and stores it in a array, and is called a noise mask. The noise mask is then added with the simulated image which brings measured image like noise in it.

2.3.1 Extraction of Noise Mask

First we define the kernel size and the stride value for its travel. In the example shown in Fig. 8 the kernel size is 3 and the stride is 1. This creates a 2D array of the size same as that of the measured image which will store the extracted noise. Make the kernel travel through the measured image keeping the stride distance between consecutive steps. For each kernel position on the image, the mean of the kernel array is subtracted from each array value, this modified array captures the noise in that image location and is stored at the same location in the noise mask array. For the overlapping regions of the kernel, the values in the noise masks are added with the new kernel values. Once the kernel finishes its travel in the measured image, the noise mask corresponding to the entire measured image is obtained.

2.3.2 Application of Noise Mask on Simulated Image

The noise mask extracted from a measured image with a particular orientation of the component will produce the best effect when applied on a simulated image with a similar component orientation. The noise mask array is directly added to the simulated image array. Finally the array values lying outside the 16 bit range (0 to 65535) are replaced with the maximum or minimum value.

2.4 ANN-ADR

The Mask R-CNN [33] model is used for the defect detection and segmentation purpose. Figure 9 shows the framework of mask R-CNN model. The Mask R-CNN framework is built on top of Faster R-CNN [26]. So, for a given image, Mask R-CNN, in addition to the class label and bounding box coordinates for each object, will also return the object mask. The defect detection system is composed of four modules. The first module is a ResNet-51 feature extractor that generates a high-level featurized representation of the input image. The second module is a CNN that proposes regions of interest (RoIs) in the image, based on the output of the feature extractor module. The third module is a CNN that attempts to classify the objects in each RoI. The fourth module performs image segmentation, with the goal of generating a binary mask for each region. Each module is described in detail throughout the remainder of this section.

The first module in the ANN system transforms the image pixels into a high-level featurized representation. The model uses ResNet-51 [34] feature extractor which is a very deep convolutional neural network with 51 trainable layers and approximately 27 million parameters. Hence it is unlikely that the network can be trained to extract meaningful features from input images, using the relatively small measured + simulated dataset. One interesting property of CNN-based feature extractors is that the features they generate often transfer well across different image processing tasks. This property is leveraged when training the proposed defect detection system by using weights pretrained on MS COCO dataset [35] for feature extractor then applying transfer learning technique to fine tuning the weights by further training the model on our custom dataset.

The second module is the region proposal network (RPN). The RPN takes a feature map of any size as input and outputs a set of rectangular object proposals, each with a score describing the likelihood that the region contains an object. The RPN is a small CNN which convolves with the output of the feature extractor. At a higher level the RPN output's a vector describing the bounding box coordinates and the likeliness of object in it at the current sliding position. Transfer learning has an important application in the training of RPN, When training an object detection network on a large dataset with many classes, the RPN learns to identify subsections of the image that likely contain an object without discriminating by object class.

The third module is the region based detector (RBD). Till now the ANN model is able to select a fixed number of regions from the original image. The region based detector classify the defects in each region, and fine-tune the bounding box coordinates. The input to the RBD is cropped from the output of ResNet-51 feature extractor, according to the shape of the regressed bounding box. The size of the input is dependent on the size of the bounding box. To address this issue, an RoIAlign layer is used to convert the input to a fixed-length feature vector. Each feature vector from the RoIAlign layer is fed into a sequence of convolutional and fully connected layers. The last fully connected layer produces two output vectors: The first vector contains probability estimates for each of the K object classes plus a catch-all background class. The second vector encodes refined bounding-box positions for one of the K classes.

The fourth module is the mask prediction module which performs the instance segmentation by predicting a segmentation mask for each RoI. The prediction of segmentation masks is performed using another CNN, referred to as the

Fig. 10 The simulated image and its annotation mask used for the training of the ANN model are shown. The annotation mask is a binary image with black background and white color defects

Simulated Rear Propellant Grain

Annotation mask
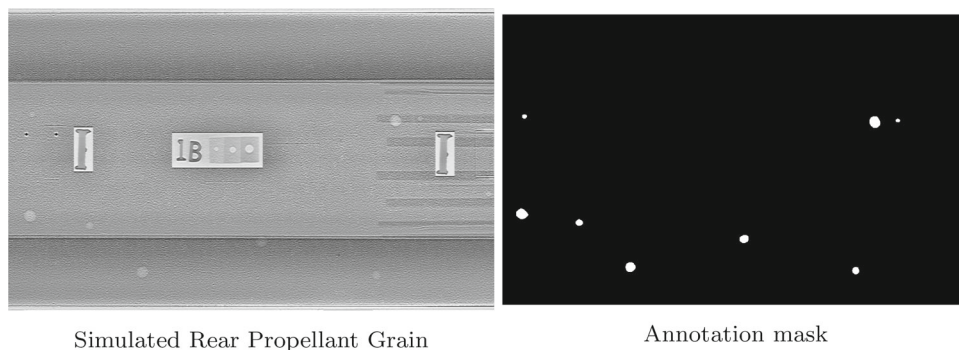
instance segmentation network. The instance segmentation network has a $28 \times 28 \times K$ dimensional output for each RoI, which encodes K binary masks of resolution 28×28, one for each of the k classes. The reader is referred to [24] for a more detailed description of the Mask R-CNN architecture.

We have used the mask R-CNN implementation from the GitHub repository [36] and trained it on our data set. We applied transfer learning on the model. The pretrained weights are used corresponding to MS COCO dataset and further trained the final layers of the model on the dataset which is a combination of experimental and simulated dataset. The dataset contains in total 2674 training images, out of which 450 are the experimental images and 2224 are simulated images. The image resolution of both the measured and simulated images are of similar size. Further to avoid the time required to simulate all the 2224 images, which is around 30 minutes per image, data augmentation techniques is used such as flipping and rotating of images and the annotation masks to multiply our simulated images. The binary annotation mask is also provided as output by the SIMXRAY corresponding to each radiograph with defects marked as white and the background is marked as black as shown in Fig:10. The tight fitting bounding box for each defect is extracted from these binary masks. The model is tested on 527 images of which 420 are simulated and 107 are experimental images.

2.5 Experimental Data Collection

A cylindrical sleeve casted composite propellant grain having a diameter of 168 mm was used for the experiment. The setup contains a 450 kV X-ray source (Make-Seifert, Germany), a flat panel detector (Make-Perkin Elmer), Digital Imaging System (Make-X-Innovation, UK) and an object manipulator having translational and rotational degree of freedom. Although the 450 kV x-ray machine was used for radiography of the solid propellant grain, but the actual energy used for the radiographic exposure is about 160 kV.

3 Results and Discussion

The automatic defect recognition for X-ray images were developed using a deep neural network based transfer learning technique. In order to overcome the small number of data set available from experiments, we used physics based data generation. Following are the results obtained from the simulation of X-ray images and the ANN based ADR system.

3.1 Simulation Results

Figure 11 presents the experimentally measured, simulated with noise mask and without noise mask x-rays along with their histograms for the front grain component. In Fig. 11a, the bottom right corner of the sample has the supporting structure present in the image. This is not modelled in the simulation. Figure 11b shows the simulated image with randomly added defects.

In radiography, histograms are an important metric to determine the quality of the x-rays. The histogram plots are plotted together with the experiment and simulated results. The effect of noise mask is clearly visible in the simulated x-ray and it's histogram. The graininess and texture in the measured image due to variety of reasons such as non-uniformity of material, manufacturing processes, component geometry, noise etc which is very difficult to match closely is very well inherited by the simulated image with noise mask. Noise mask increased the smoothness and improved the distribution of histogram. The kernel size and stride values used were 7 and 3 respectively.

Similarly, Fig. 12 presents the simulation output for rear grain of the propellant grain. The sample supporting material is also present in the case experimental data at the bottom.

Our noise modelling technique implicitly accommodates the effects of unsharpness and the scattered radiation modelling into consideration as can be seen by the improvement in the radiograph and it's histogram quality after applying the noise mask in b and c part of Figs. 11 and 12. We are very well able to match the contrast of the simulated image with the
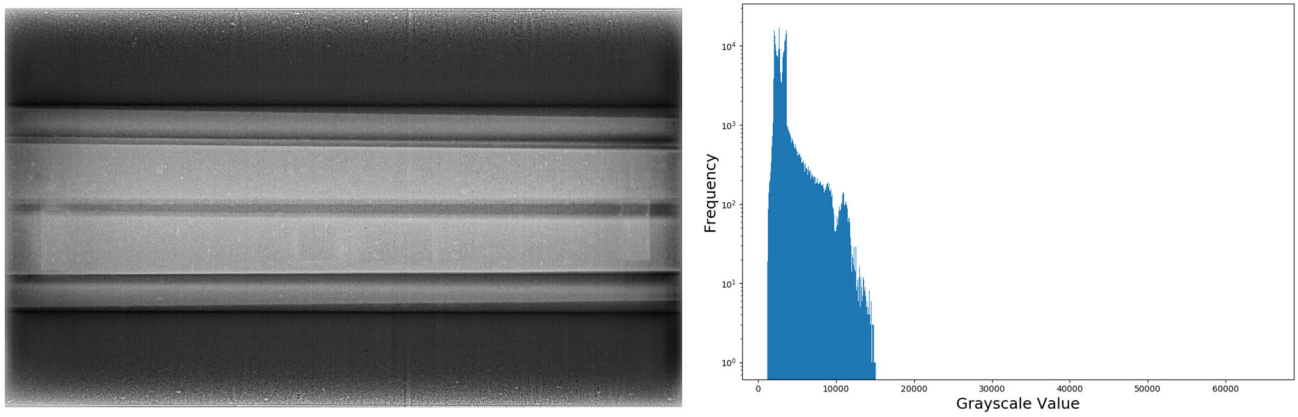
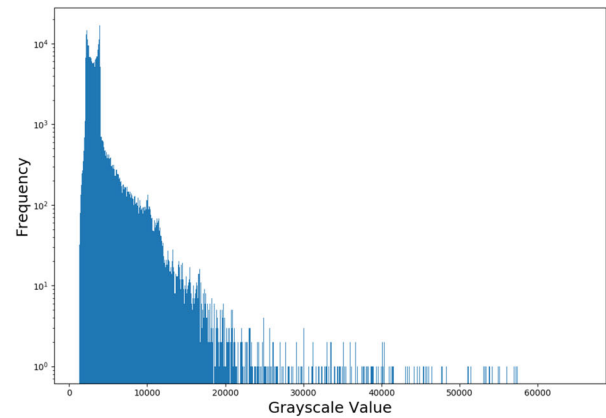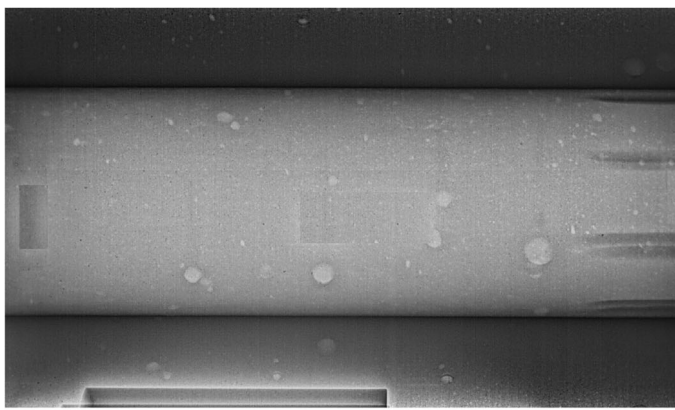Experimental X-ray.
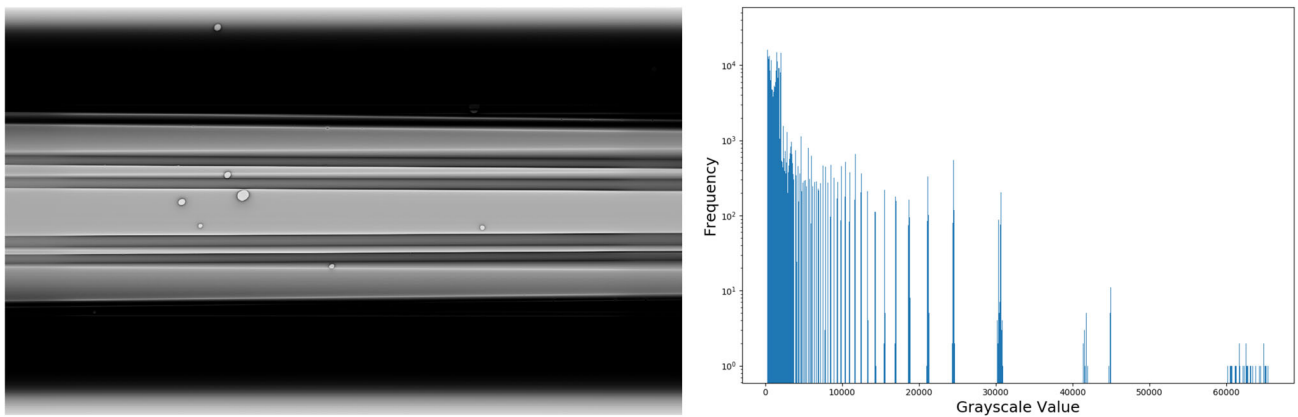
Simulated X-ray without noise mask.
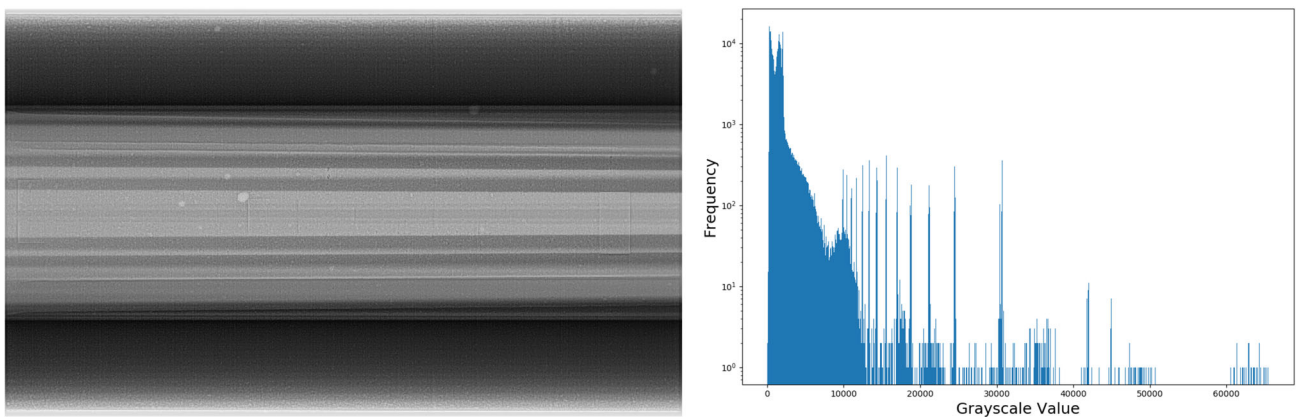
Simulated X-ray with noise mask.

Fig. 11 Comparison of Experimental, Simulated and Noise added simulated X-ray image and histograms of rear propellant grain

experimental image and also their histograms. The contribution of our noise modelling technique to model the effects of scattering and the unsharpness can be seen by comparing the defects and the edges in the simulated images without noise (b part) and with noise (c part) in Figs. 11 and 12 respectively.
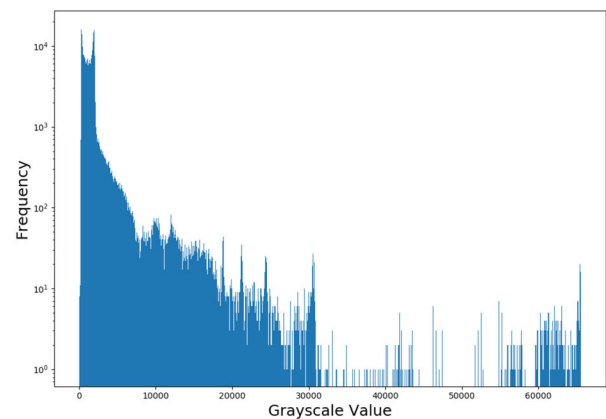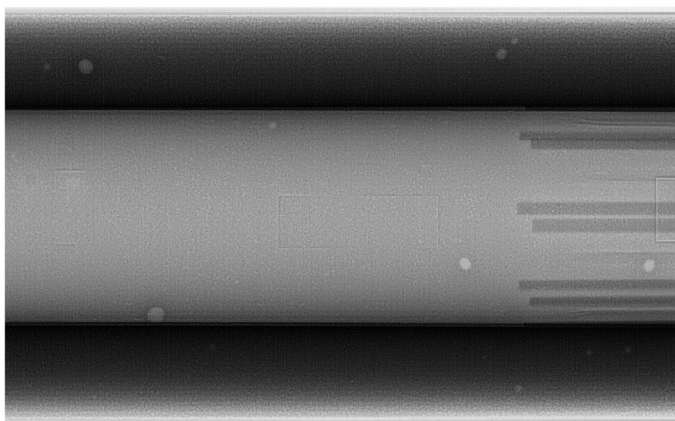
3.2 ANN-ADR Results

We configured the ANN model on our system and fine tuned the training parameters such as input image size, batch size and learning rate according to our data set and GPU capabil-

Experimental X-ray.
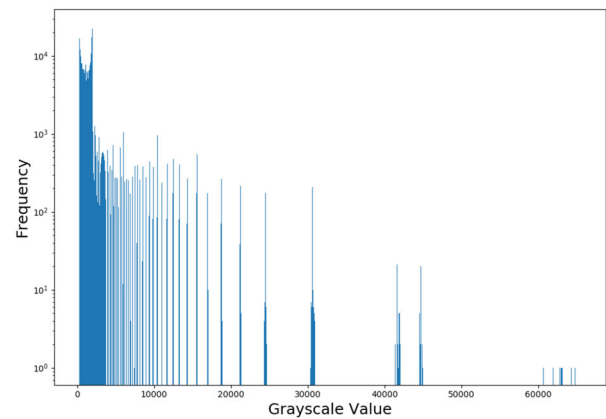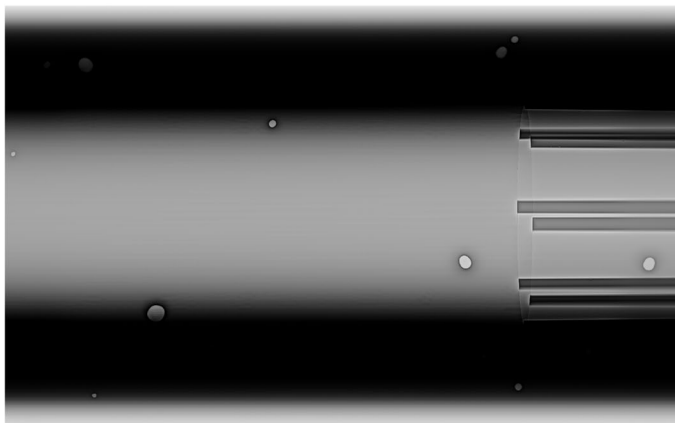
Simulated X-ray without noise mask.

Simulated X-ray with noise mask.

Fig. 12 Comparison of Experimental, Simulated and Noise added simulated X-ray image and histograms of front propellant grain

ity. The ANN training was carried out using 416 experimental images and 1754 simulated images. To see the effect of the simulated images on the ANN model accuracy we trained 5 ANN models with an increasing proportions of measured vs simulated images as given in Table 1. The trial 1 was performed using only the experimental images and no simulated images were used. In trial 2, an almost equal number of simulated images added as the experimental images into our training dataset, and in the subsequent trails we kept increas-

Table 1 ANN model runs

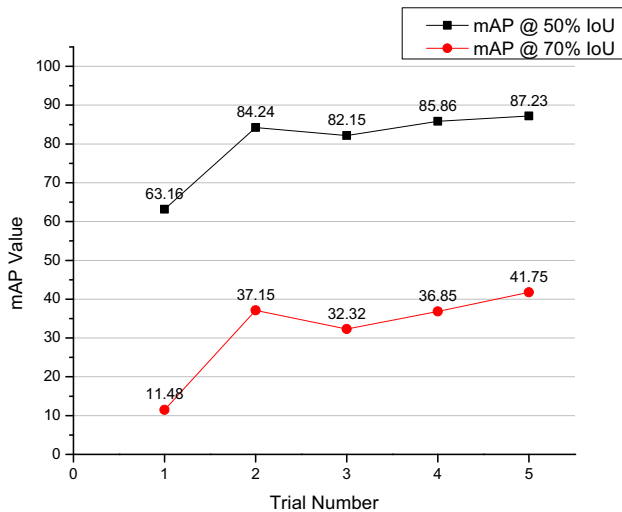| Trials	Experimental images	Simulation images	Total training images
Trial 1	416	0	416
Trial 2	416	426	842
Trial 3	416	770	1186
Trial 4	416	1198	1614
Trial 5	416	1754	2170

Fig. 13 Mean average precision (mAP) on the test set of 420 images, given different sized training sets according to Table 1. The segmentation accuracy is shown
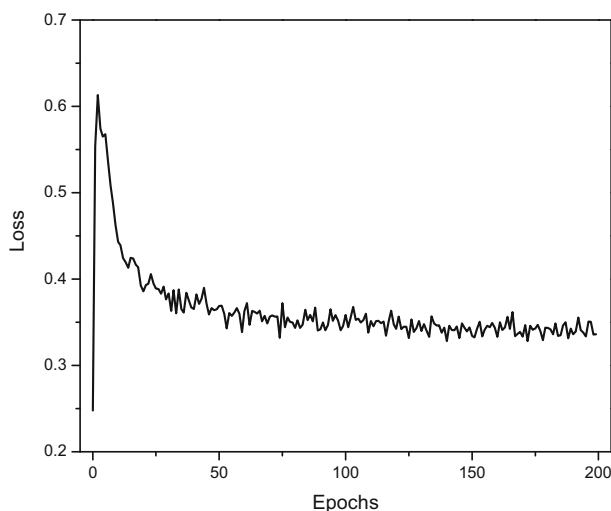

Fig. 14 Loss of the Trial 5 ANN model vs Number of Epochs

ing the number of simulated images such that in trial 5 we used all the simulated images for the training

Further, the performance of all the five trained models were evaluated on a test dataset of 420 images. We used mAP (mean Average Precision) metric at two IoU values of 50% and 75% which are the standard metrics for the object detection and segmentation problems to measure the performance of the model. The mAP values at IoU of 50% and 75% of the trials are shown in Fig. 13. As with many deep learning models, it takes large amount of labelled data to train model accurately. This exercise helps us see how the large simulated data affects the performance. The performance improved significantly in trial 2, when we doubled the training dataset with the help of simulated images. Following which the growth rate decreased. This result suggests that the chosen maskR-CNN model can learn the problem well with 842 examples (Trial 2), with quite modest improvements seen with 2170 examples (Trial 5). The performance of the neural network can continually improve as more and more data is provided to the model, but the capacity of the model must be adjusted to support the increase in data. Eventually, there will be a point of diminishing returns where more data will not provide more insight into how to best model the problem. For simpler problems, this point will be reached sooner than in more complex problems. Similar observations can also be seen in the study performed by Sun et al. [37] and Joulin et al. [38].

The maximum mAP attained is 85.23% for a 50% IoU and 41.75% for 75% IoU.

Figure 14 shows the loss calculated at each epoch and is gradually decreases and stabilizes at higher epochs. A total of 200 epochs were used. Figure 15 shows the examples of the detection of the casting defects for three different components corresponding to trial 5 ANN model. The images on the left side shows the enhanced image and on the right the ADR processed image. In all the three cases the model is able to identify and segment the void defects with in the allowed threshold limit and are marked as red pixels. Despite the inherent grainy nature of the composite, the voids with acceptable area are identified.

4 Conclusion and Future Work

Radiography is a tool commonly used by NDT for evaluating propellant casting components. Human experts are interpreting a large number of radiographic images which can lead to a subjective interpretation with the possible risk of miss-

Fig. 15 Defect detection results by Trial 5 ANN model on experimental images. Images on the left are the original images and on the right are the output of the ANN model with the detected defects shown in red segments
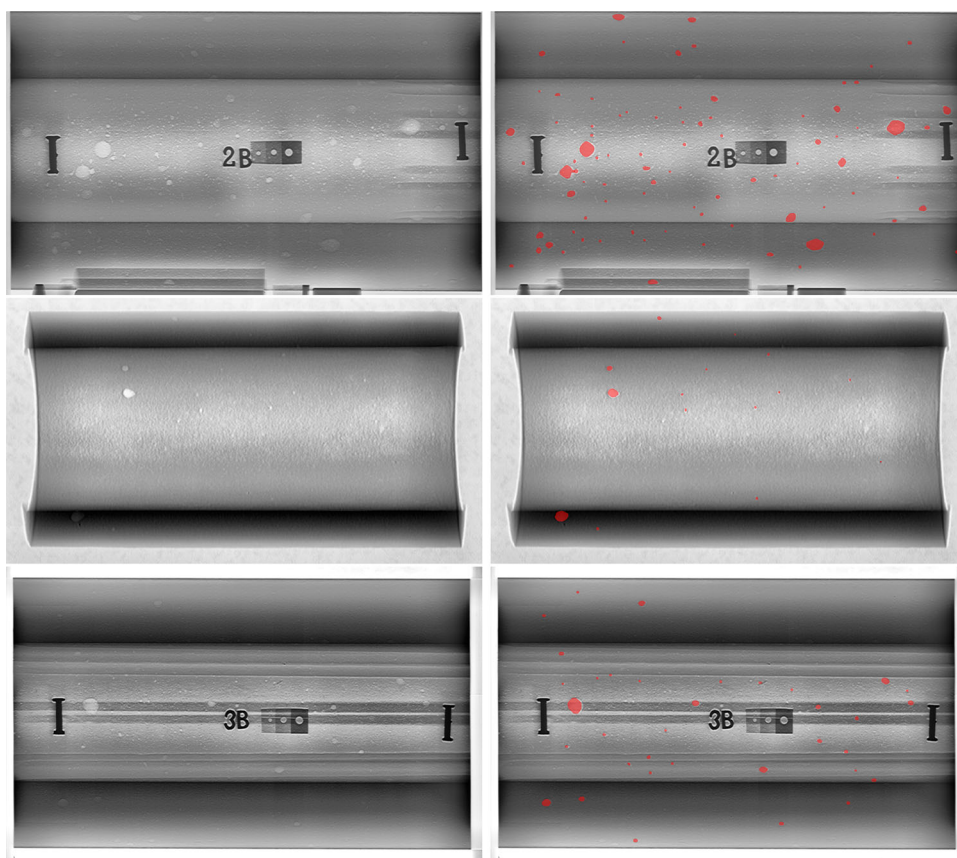
ing flaws. There are very few recorded attempts in the past to automatically detect flaws on propellant casting components. In this paper, an automatic defect recognition system for propellant casting components using transfer learning technique is demonstrated. The method used Mask R-CNN to extract the features of propellant radiographs. The limitation in the data availability is handled using data generation using radiographic simulation. A ray tracing based model is established and carefully mapped the noise in the experimental data to generate realistic radiographic images in large scale. The void type defects were inserted in the CAD geometry and generated the defect features in the radiographic images.

The ray tracing technique is used for the simulation of the industrial digital X-ray imaging system. The noise in the experimental images are extracted and incorporated in the simulated images in order to use these images for the ANN based ADR system. The simulated images are in good agreement with the experimental results.

The experimental data combined with various amount of simulated data is then used to train mask R-CNN model which produces the masks of the defects in the image. Our work shows that there is a significant improvement in the ANN model accuracy when the model is trained with large amount of simulated data combined with experimental data compared to when the model is trained only on experimen-

tal data. We are able to achieve an improvement of 24% in the $mAP@50\%IoU$ and an improvement of 30% in the $mAP@75\%IoU$ metrics. This approach can be easily used in the propellant manufacturing line for real time evaluation.

Our approach is capable of simulating radographs of propellant grains with a wider variety of defects, having different material density such as cracks, slags, agglomerates and inclusions which occurs during manufacturing. We are planning to do the classification problem as part of our future work. In the future, we will extend the sample sets from both experiments and simulation. The current generation time for the X-ray images are high and we are planning to incorporate GPU based acceleration for this process. The data generation using Deep Convolutional Generative Adversarial Network (DCGAN) is also planned as a future task.

References

1. Brooks, W.: Solid propellant grain design and internal ballistics, vol 8076. National Aeronautics and Space Administration (1972)

2. Davenas, A.: Solid Rocket Propulsion Technology. Newnes, Oxford (2012)
3. Halmshaw, R.: Industrial Radiology: Theory and Practice, vol. 21. Chapman Hall, London (2012)
4. Ghose, B., Kankane, D.: Estimation of location of defects in propellant grain by X-ray radiography. NDT E Int. **41**(2), 125–128 (2008)
5. Deprins, E.: Digital radiography in ndt applications. CINDE J. **25**(4), 14–16 (2004)
6. Ravindran, V.: Digital radiography using flat panel detector for the non-destructive evaluation of space vehicle components. Journal of Non-Destructive Testing & Evaluation **4**(2), (2005)
7. Willems, P., et al.: Image quality comparison for digital radiography systems for ndt. In: 15th WCNDT (2000)
8. Godoi, W., da Silva, R., Swinka-Filho, V.: Pattern recognition in the automatic inspection of flaws in polymeric insulators. Insight Non-Destruct. Test. Cond. Monitor. **47**(10), 608–614 (2005)
9. Liao, T.W., Li, Y.: An automated radiographic ndt system for weld inspection: part iiflaw detection. Ndt E Int. **31**(3), 183–192 (1998)
10. Wang, G., Liao, T.W.: Automatic identification of different types of welding defects in radiographic images. Ndt E Int. **35**(8), 519–528 (2002)
11. Da Silva, R.R., Siqueira, M.H., de Souza, M.P.V., Rebello, J.M., Calôba, L.P.: Estimated accuracy of classification of defects detected in welded joints by radiographic tests. NDT E Int. **38**(5), 335–343 (2005)
12. Saravanan, T., Bagavathiappan, S., Philip, J., Jayakumar, T., Raj, B.: Segmentation of defects from radiography images by the histogram concavity threshold method. Insight Non-Destruct. Test. Cond. Monitor. **49**(10), 578–584 (2007)
13. Da Silva, R.R., Mery, D.: Radiographic testing: part l image processing. Mater. Eval. **643**, 1–9 (2007)
14. Zahran, O., Kasban, H., El-Kordy, M., Abd El-Samie, F.: Automatic weld defect identification from radiographic images. Ndt E Int. **57**, 26–35 (2013)
15. Lashkia, V.: Automatic weld defect identification from radiographic images. Image Vis. Comput. **19**(5), 261–269 (2001)
16. Da Silva, R., Siqueira, M.H., Caloba, L.P., Rebello, J.: Radiographics pattern recognition of welding defects using linear classifiers. Insight **43**(10), 669–74 (2001)
17. Da Silva, R., Calôba, L., Siqueira, M., Sagrilo, L., Rebello, J.: Evaluation of the relevant characteristic parameters of welding defects and probability of correct classification using linear classifiers. Insight **44**(10), 616–22 (2002)
18. Liao, T.W.: Improving the accuracy of computer-aided radiographic weld inspection by feature selection. Ndt E Int. **42**(4), 229–239 (2009)
19. Valavanis, I., Kosmopoulos, D.: Multiclass defect detection and classification in weld radiographic images using geometric and texture features. Expert Syst. Appl. **37**(12), 7606–7614 (2010)
20. Zapata, J., Vilar, R., Ruiz, R.: Performance evaluation of an automatic inspection system of weld defects in radiographic images based on neuro-classifiers. Expert Syst. Appl. **38**(7), 8812–8824 (2011)
21. Boaretto, N., Centeno, T.M.: Automated detection of welding defects in pipelines from radiographic images dwdi. Ndt E Int. **86**, 7–13 (2017)
22. Baniukiewicz, P.: Automated defect recognition and identification in digital radiography. J. Nondestruct. Eval. **33**(3), 327–334 (2014)
23. Zapata, J., Vilar, R., Ruiz, R.: Automatic inspection system of welding radiographic images based on ann under a regularisation process. J. Nondestruct. Eval. **31**(1), 34–45 (2012)
24. Ferguson, M.K., Ronay, A., Lee, Y.T.T., Law, K.H.: Detection and segmentation of manufacturing defects with convolutional neural networks and transfer learning. Smart Sustain. Manuf. Syst. **2**, 10 (2018)
25. Mery, D., Riffo, V., Zscherpel, U., Mondragón, G., Lillo, I., Zuccar, I., Lobel, H., Carrasco, M.: Gdxray: The database of X-ray images for nondestructive testing. J. Nondestruct. Eval. **34**(4), 42 (2015)
26. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
27. Duvauchelle, P., Freud, N., Kaftandjian, V., Babot, D.: A computer code to simulate x-ray imaging techniques. Nucl. Instrum. Methods Phys. Res. Sect. B **170**(1–2), 245–258 (2000)
28. Lazos, D., Bliznakova, K., Kolitsi, Z., Pallikarakis, N.: An integrated research tool for X-ray imaging simulation. Comput. Methods Progr. Biomed. **70**(3), 241–251 (2003)
29. Fanti, V., Marzeddu, R., Massazza, G., Randaccio, P.: A simulation tool to support teaching and learning the operation of X-ray imaging systems. Med. Eng. Phys. **27**(7), 555–559 (2005)
30. Bellon, C., Jaenisch, G.R.: Artist-analytical rt inspection simulation tool. In: Proc DIR, pp. 25–27 (2007)
31. Tucker, D., Barnes, G., Chakraborty, D.: Semiempirical model for generating tungsten target X-ray spectra. Med. Phys. **18**, 211–8 (1991a). https://doi.org/10.1118/1.596709
32. Tucker, D.M., Barnes, G.T., Wu, X.: Molybdenum target X-ray spectra: a semiempirical model. Med. Phys. **18**(3), 402–407 (1991b). https://doi.org/10.1118/1.596686
33. He, K., Gkioxari, G., Dollr, P., Girshick, R.: Mask r-cnn. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988 (2017)
34. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition, pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
35. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision - ECCV 2014, pp. 740–755. Springer, Cham (2014)
36. Abdulla, W.: Mask r-cnn for object detection and instance segmentation on keras and tensorflow (2017). https://github.com/matterport/Mask_RCNN
37. Sun, C., Shrivastava, A., Singh, S., Gupta, A.: Revisiting unreasonable effectiveness of data in deep learning era (2017)
38. Joulin, A., van der Maaten, L., Jabri, A., Vasilache, N.: Learning visual features from large weakly supervised data (2015)