

## ARTICLE OPEN

## Automated defect analysis in electron microscopic images

Wei Li<sup>1</sup>, Kevin G. Field<sup>2</sup> and Dane Morgan<sup>1</sup>

Electron microscopy and defect analysis are a cornerstone of materials science, as they offer detailed insights on the microstructure and performance of a wide range of materials and material systems. Building a robust and flexible platform for automated defect recognition and classification in electron microscopy will result in the completion of analysis orders of magnitude faster after images are recorded, or even online during image acquisition. Automated analysis has the potential to be significantly more efficient, accurate, and repeatable than human analysis, and it can scale with the increasingly important methods of automated data generation. Herein, an automated recognition tool is developed based on a computer vision-based approach; it sequentially applies a cascade object detector, convolutional neural network, and local image analysis methods. We demonstrate that the automated tool performs as well as or better than manual human detection in terms of recall and precision and achieves quantitative image/defect analysis metrics close to the human average. The proposed approach works for images of varying contrast, brightness, and magnification. These promising results suggest that this and similar approaches are worth exploring for detecting multiple defect types and have the potential to locate, classify, and measure quantitative features for a range of defect types, materials, and electron microscopic techniques.

*npj Computational Materials* (2018)4:36; doi:10.1038/s41524-018-0093-8

## INTRODUCTION

Electron microscopy is one of the most important methods for studying the structural and morphological properties of materials from the micrometer to the angstrom scale. Electron microscopic images provide researchers rich information on both repeated structural units (e.g., unit cells of crystals) and defected regions (e.g., grain boundary, impurities, defect clusters) by showing the responses of electrons interacting with the material. In this work, we focused on a particular but widely used application of electron microscopy, which is analyzing the locations and sizes of defects of metal alloys under irradiation.<sup>1</sup> While we focused on defects from radiation damage, the tools developed herein could be readily modified to assess many types of features in microscopic images, from counting nanoparticles to identifying line dislocations.

Irradiation damage in materials for nuclear applications greatly affects the durability of existing nuclear reactor facilities and advanced reactor designs. Understanding the effects of irradiation on materials properties and performance is critical to safe and reliable nuclear reactor operation. Study of irradiation damage processes and mechanisms, as well as testing of new nuclear materials, requires several series of experiments with irradiated materials and repetitive data generation using electron microscopy. The repetitive generation of microstructure images is needed to obtain statistically significant information on the total number and distribution of different types of preexisting and radiation-induced/-enhanced defects, such as grain boundaries, precipitates, dislocation lines and loops, stacking fault tetrahedral, voids, bubbles, and "black-spot" defects.<sup>1,2</sup>

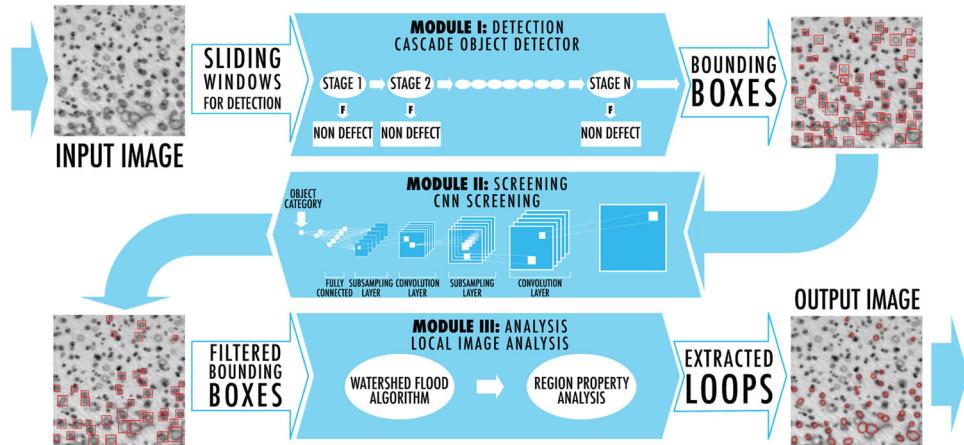
Manually identifying defects and counting the relevant properties to obtain statistically accurate values has four major issues. (1) Manual identification can be time-consuming, especially for a large data set of microscopy images (e.g., >25–50 images),

requiring typically anywhere from just a few minutes to about an hour per image. (2) Manual identification is error prone, as it is easy to miss a defect or identify one incorrectly. (3) Manual identification lacks consistency and reproducibility. Analysis can be significantly impacted by the human bias and training of a given researcher, making it difficult to compare results across groups or determine absolute behavior. (4) Manual identification does not scale well. New high-speed detectors have enabled electron microscopes to take from tens up to thousands of images per second,<sup>3</sup> and automated sample exploration might generate data sets with thousands or millions of images. Even at just minutes per image, automated analysis is essential to take advantage of these emerging and inevitably increasing data generation capabilities. The development of image analysis tool for defect detection could reduce the time required for analysis to almost zero; provide accurate, consistent, and unbiased results; and scale approximately linearly with available computational resources.

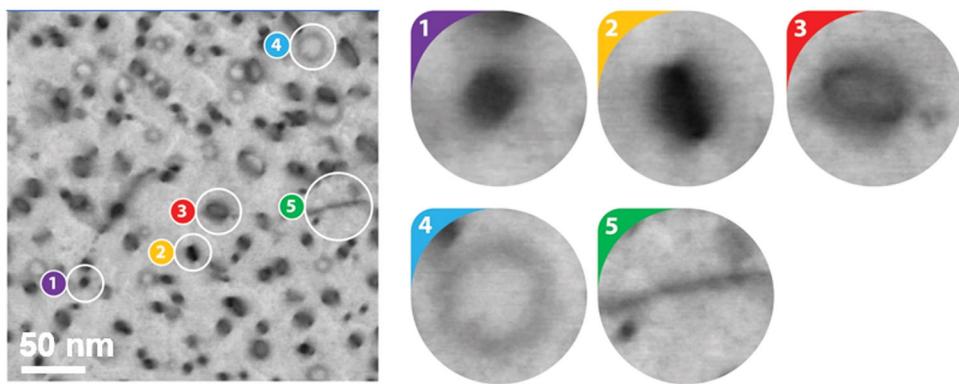
Early image recognition system(s) mainly used scale-invariant feature transform,<sup>4</sup> binary feature histogram,<sup>5</sup> and histogram of oriented gradients (HOGs)<sup>6</sup> to extract the image features and then entered these image features into a classifier. In recent years, there has been continuing development of the computer vision field and a series of successes in achieving human accuracy in various image recognition tasks.<sup>7–12</sup> One of the more successful approaches was developed by Paul Viola and Michael Jones, who proposed a fast object detection structure known as the "cascade object detector for face detection" application.<sup>8,13</sup> More recently, convolutional neural networks (CNNs)<sup>14</sup> have emerged as a very powerful approach to image recognition tasks.<sup>15–17</sup> A CNN—a kind of artificial neural network that operates convolution directly on raw pixel intensity data—consists of several repeating layers of convolution, nonlinearity, and pooling, followed by fully

<sup>1</sup>Department of Materials Science and Engineering, University of Wisconsin–Madison, Madison, WI 53706, USA and <sup>2</sup>Oak Ridge National Laboratory, Oak Ridge, TN 37830, USA  
Correspondence: Dane Morgan (ddmorgan@wisc.edu)

Received: 26 February 2018 Revised: 15 June 2018 Accepted: 18 June 2018  
Published online: 18 July 2018



**Fig. 1** Schematic flow chart of the proposed automated detection approach. Input micrographic images go through the pipeline of module I—Cascade Object Detector, module II—CNN Screening, and module III—Local Image Analysis. After module I, the loop locations and bounding boxes are identified and then further refined to remove false positives using module II. Then module III determines the loop shape and size



**Fig. 2** Selected bright-field scanning transmission electron microscopy (STEM) image of ferritic alloys showing the common defect types: (1, 2) closed circular/elliptical solid loops, (3) open ellipse loops, (4) open circular loops, and (5) line dislocation segments. Open ellipse loops (3) were selected for automated detection. Image size:  $1024 \times 1024$  pixels; insets scaled arbitrarily

connected layers.<sup>18</sup> CNN is a rapidly developing field, and new CNN structures are still being proposed.<sup>17,19–23</sup>

Recent advances in computer vision and machine learning (ML) have been introduced into the electron microscopic field for image analysis, such as detection and segmentation in medical images,<sup>24–26</sup> unsupervised statistical representation of microstructure images,<sup>27</sup> clustering or classification in various materials images,<sup>28–30</sup> chemical identification and local transformation tracking at the atomic level in scanning transmission electron microscopic (STEM) images,<sup>31</sup> and analysis of electron diffraction patterns.<sup>32</sup> To our knowledge, there is very limited published research work on using computer vision or deep learning approaches to recognize locations and extract quantitative information for nanoscale and mesoscale defects in microscopic images. Ziatdinov et al.<sup>31</sup> recently reported their work on using deep learning to detect location of the atomic species and type of lattice defects for atomically resolved images, but their approach did not detect defects at a larger scale ( $>1$  nm) or extract quantitative shape or contour information of the defects. There exist two main challenges to detecting defect structures in microscopic images: (1) the lack of sufficient annotated micrographs for training and (2) the difficulty of extracting an accurate defect contour in an automated way. With respect to the first issue, we have used a fairly large manually labeled database combined with image augmentation approaches, but we note that direct simulation of TEM images could provide much larger

data sets, although they would not be from physical material measurements. With respect to the second issue, existing image analysis tools, like edge extraction and/or Hough transform, often involve manual adjustments of image contrast or brightness, multiple parameter inputs, and slow image processing speeds. Current ML approaches recognize objects in the image accurately but do not have mechanisms for extracting quantitative defect structure and defect distribution information, such as defect dimensions and areal number density.

Herein an approach is developed for automated defect detection and analysis that breaks the task into three stages. We applied different computer vision techniques in each stage and thereby achieved the combined advantages of several methods. Figure 1 portrays the essential steps involved in the automated detection workflow: (1) detection module I with cascade object detector; (2) screening module II with CNN; (3) and analysis module III with two local image analysis methods—a watershed flood algorithm to find the defect contour and a region property analysis to obtain size information for the defect contour.

## RESULTS

Microstructural images of irradiated steels typically contain several prominent types of defects: open circular/elliptical loops with matrices inside, closed circular/elliptical solid loops, and line dislocations. Micrographic images can have various contrast,

brightness, and degrees of focus (sharpness) and sometimes contain different types of defects that might distract from one another. Figure 2 shows a sample micrographic image obtained from a ferritic alloy irradiated in materials test reactor to introduce various defect sizes and number densities. Among all types of defects in electron microscopic images, open elliptical loop defect detection, inset 3 in Fig. 2, is a particularly challenging task. Here we define an open elliptical loop, which is a  $a/2\langle 111 \rangle$  dislocation loop for the studied material, as a defect with white–black–white contrast across a loop segment and shows notable reduced pixel intensity within the interior of the defect. A perfect closed ellipse is not essential, partial loops are also defined as open ellipse loops as these truncate the free surface of the specimen. At reduced loop sizes—typically  $<10$  nm, the distinction between open elliptical loops and black dots, inset 1 in Fig. 2, becomes ambiguous and open to interpretation. Here, as discussed in detail later, we use the expertise of two seasoned researchers to limit ambiguous distinctions in the definition of open elliptical loops within the training data sets. Open circular loops that are in-plane  $\langle 100 \rangle$  loops, inset 4 in Fig. 2, are easily distinguishable as they exhibit distinct white–black–white–black–white contrast across a singular section of the loop.

Open elliptical loops have the disadvantages of limited effective information (only pixels on or near the loop contour are useful pixels), heavy background noise, blurred patterns, and distracting patterns (line dislocations, open circular loops, and closed loops). Therefore, in demonstrating the effectiveness of the automated defect analysis approach on open elliptical loop defects, we are demonstrating it on one of the most difficult defect detection problems in irradiated materials, suggesting the method is likely to also work on other defect types.

### Strategy

Figure 1 shows the steps that are crucial for successfully extracting and characterizing all the open ellipse-like loop defects in the micrographic images in Fig. 2, organized by modules. In module I, we trained a cascade object detector for defect recognition to enable it to accurately determine bounding boxes that contain a single loop defect inside. The detector located almost all the defect positions in the image but was specifically tuned to have a high false detection error. Tuning for lower false detection errors caused the detector to miss larger numbers of real loops within the image. A screening method was added in module II to effectively decrease the false detection rate. The defect screening method was fundamentally a binary image classification problem based on the features of single defect images. The screening method consisted of a 15-layer CNN, including an image input layer, three convolutional layers, four nonlinearity layers, and three pooling layers with two fully connected layers, one softmax layer and one class-output layer (see Methods section for more information.).

After running modules I and II, we obtained a list of bounding boxes that each ideally contained one loop. To extract the loop shape from the detected bounding boxes in module III, we adapted a watershed flooding algorithm from the algorithm introduced by Luc Vincent and Pierre Soille.<sup>33</sup> The watershed algorithm is an image segmentation method based on an immersion process analogy, in which the flooding of the water in a picture is efficiently simulated using a queue of pixels. We implemented our own watershed flood algorithm specifically for microstructure loop defects and achieved high accuracy for extracting defect edges, regardless of the original image brightness and contrast. We also included region property analysis, using the Matlab regionprops tool,<sup>34</sup> in module III to fit the extracted loop shape with an ellipse to obtain the shape metrics of loops, such as length of the long axis and orientation of the loop. Through the pipeline of all three modules, we managed to identify

loops and extract the interested loop shape information in an automated way with minimal tunable parameters.

### Data preparation and augmentation

We collected a data set of 298 micrographic images with a human count of 9566 total loop defects, which consisted of several ferritic alloys irradiated under varying experimental conditions to introduce various defect sizes and number densities. All images were generated using the on-zone STEM imaging technique, which reduces background variations and increases signal-to-noise ratios<sup>35</sup> (see Methods section for more information). The data set was split into a training set and a test set at a fixed 9:1 ratio. The test set was selected at random from the complete image library. Note that there is no method of judging absolutely rigorously whether a pattern is a loop, and different researchers can disagree regarding ambiguous patterns. However, to assess performance we must set some “ground truth” labeling as the true correct labeling. Therefore, both the training and test data sets were manually labeled by two researchers experienced in loop labeling who worked together; and the labeling in the test data set was revisited multiple times by the two researchers in an attempt to ensure correct labels. The agreed-upon labeling from these two researchers served as the ground truth for testing the performance of the automated detection model. After data set splitting, we obtained a training set of 270 images and test set of 28 images, containing 8424 and 1142 human-identified loops, respectively. The training data set of micrograph images was augmented to a total of 1605 images (39,596 human-identified loops) using rotating and mirroring operations on images. Image augmentation can provide more training instances for the model and make the model invariant to rotated or flipped image inputs. The image augmentation improved the overall performance of the cascade object detector by both adding more training examples and enabling the cascade object detector to use more stages.<sup>18,36</sup>

We also collected a data set of about 500 micrographic images without loop defects to serve as negative images for the computer vision model. Images were either collected using the same electron microscopic technique as the positive images or generated from positive images on which loop areas were covered by patches. The patches were cropped and rotated/flipped from nearby regions around the loop areas. The function of negative images is to provide negative instances to train the model in identifying non-loop regions. The negative images were augmented to  $>15,000$  images by rotating, resizing, and mirroring operations.

### Evaluation metrics on detection accuracy

To evaluate the performance of the automated defect analysis approach, as well as the accuracy of each module chained in the approach, we used two sets of evaluation metrics: (1) recall and precision metrics<sup>37</sup> to define the capability to find correct positions of the loop defects inside the image; and (2) defect size distribution metrics, including the total number of defects in the image, the average diameter (length of the major axis) of the loop defects in the image, and the standard deviation of the diameters of the loop defects in the image. These metrics were used to define the capability to correctly gather statistical information on loops in the image, which will eventually be used for materials characterization and analysis. Recall is defined as the percentage of correct predictions in all human-labeled loop positions. Precision is defined as the percentage of correct predictions in all the machine-labeled positions. A loop position was given according to the center of the loop bounding box, and agreement was considered to be obtained if two values were  $\leq 20$  pixels apart in both the X and Y directions. This value allows for the fact that there is some uncertainty in loop positions of the ground truth. The value is about 2% in each direction of the total image

**Table 1.** Summary of recognition accuracy of loop position evaluated using precision and recall metrics

Module	Test set of 28 images (complete)		Test set of 6 images (selected)	
	Recall	Precision	Recall	Precision
Module I	0.904	0.639	—	—
Modules I and II	0.858	0.865	0.842	0.837
Human average	—	—	0.804	0.790

The test set of 28 images indicates the complete test data set, and the test set of 6 images consists of 6 representative images selected from the complete test set that are used for comparison of machine labeling and human labeling. "Human average" represents the average performance of five researchers (discussed later in this section)

grid sizes of  $1024 \times 1024$  pixels. Twenty pixels roughly equated to 2.84–9.48 nm within the images, so this metric required nanometer precision for the loop predictions. Overall, recall measures how well the machine can avoid missing human-labeled loops, and precision measures how well the machine can avoid giving false identifications (marking a non-loop pattern as a loop defect). Because modules I and II provided the loop positions as output, they were each evaluated by the first set of recall and precision metrics. Because module III provided the actual defect size distributions, it was evaluated using the second set of defect size distribution metrics. The overall performance of the automated defect analysis approach was the combination of the recall and the precision from the combined modules I and II and the image analysis results from module III.

#### Cascade object detector

The cascade object detector used cascading to assemble multiple stages of classifiers, using all information collected from the output from a given classifier as additional information for the next classifier in the cascade. The training of the cascade object detector involved feeding the detector with a set of images with loops and the corresponding bounding box coordinates, as well as a set of images with no loops as negative examples. After the detector is trained, it can be applied to an image and detect the loops in the image. To search for loop defects in the entire frame, search windows of various sizes can be moved across the image and check every location for the detector. The examined windows that successfully pass through all stages in the cascade object detector are then output as the detected loops in the image.

A 40-stage cascade object detector was trained on the augmented image training data set of 1605 images. After training, the cascade object detector was applied to the non-augmented test image set of 28 images. The average recall of all the images in the test set was 0.904, and the average precision of all the test images was 0.639. These results indicated that, on average, 90% of the loops in each image were detected, and 64% of the detected loops were also identified as loops by manual labeling by a human researcher. These data are included in the summary shown in Table 1.

#### CNN screening

A 15-layer CNN was added after the cascade object detector. The CNN screening improved the precision of the loop recognition and added the flexibility of tuning the precision and recall score of the detector. To train a CNN for loop/non-loop image classification, we constructed a CNN training set of cropped images with/without a loop inside by applying the cascade object detector on the augmented training data set. We built a CNN consisting of an input layer, three convolution layers, four nonlinearity layers, three

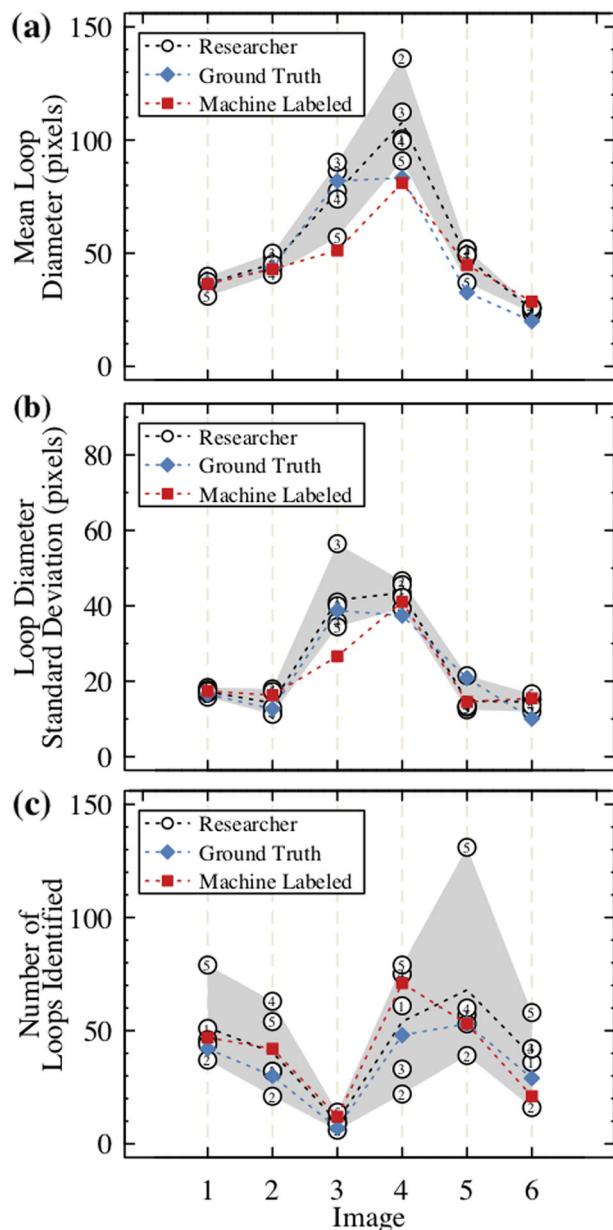
maximum pooling layers, two fully connected layers, one softmax layer, and one class-output layer. The convolution layer is a feature extraction layer that uses various kernels to convolve the whole image, as well as the former feature map to get new features. Nonlinearity layers decide how the extracted features can be passed to the next layer, and pooling layers reduce the feature dimension by down-sampling the features extracted by convolutional layers. The unique advantage of CNN compared with other machine-learning-based methods lies in its effective feature extraction due to sparse connectivity and shared local weights, which both reduces the memory requirements of the model and improves its statistical efficiency.<sup>18</sup>

To test the effectiveness of the CNN, we first collected all the bounding boxes produced by the cascade object detector on the test image set. Each bounding box was considered to contain one loop defect, as predicted by the detector. Each candidate box was then fed into the CNN to determine whether it was a true loop or false loop. The CNN screening greatly improved the performance of the cascade object detector by adding an additional screening stage with a more powerful self-learned feature representation compared with the local binary pattern (LBP) used by the cascade object detector. The chained approach using the cascade object detector and CNN on the 28 test images resulted in an average recall of 0.858 and an average precision of 0.865. These results are summarized in Table 1. With CNN screening added to the cascade object detector, we improved the average precision from 0.639 to 0.865, with only a small sacrifice in average recall from 0.904 to 0.858.

#### Defect extraction and comparison with human average

After loop detection and screening with modules I and II, in module III, we extracted the loop edge and shape information using a watershed flood algorithm and region property analysis,<sup>34</sup> respectively. The watershed flood algorithm was modified in this work for loop edge extraction and was intended to work for every image with a single loop inside, regardless of image-to-image variations in brightness, contrast, and noise level. The modified flood algorithm worked in four steps: (1) find the centroid position of the loop and set it as the inside region; (2) set the edges of the bounding box as the outside region; (3) gradually and separately increase the pixel intensity value threshold and expand the inside and outside regions by adding neighboring pixels with pixel intensity below the threshold; and (4) mark the meeting point of the inside and outside regions as the loop edge. For region property analysis, we used connected component analysis (grouping image pixels into components based on pixel connectivity)<sup>38</sup> to obtain the loop region and measure loop shape information such as centroid point and major/minor axis length, as well as the angle between the major axis and the image x axis. The region property functions "regionprops"<sup>34</sup> were used directly from the image processing toolbox in Matlab for the connected component analysis.

To evaluate the accuracy of loop shape information results, we compared the human analysis result and machine recognition result for loop density, mean major axis length of loop defects, and standard deviation of the major axis length. To assess the impact of ambiguous defects that might be either loops or not loops and of human labeling mistakes, we selected six representative images from the test image set and collected the labels assigned to the six images by five seasoned experts (>5 years in the field) working in the field of characterization of radiation damage. All five researchers were given standard training in loop labeling, using the same common image analysis software for electron microscopic images (ImageJ<sup>39</sup>) through the same tutorial video (see Supplementary Information Section II). For the defect size distribution metrics comparison, we collected the number of loops, mean major axis length, and standard deviation of major



**Fig. 3** Comparison of **a** mean loop diameter, **b** standard deviation of mean loop diameter, and **c** number of loops derived from both the human and machine labeling. Data are from domain expert researchers (open circles), the ground truth labeling developed by the authors (blue diamonds), and the automated machine labeling (red rectangles). The image number corresponds to the image number in Fig. 4.

axis from all the research labeling and the machine labeling for each image. For the recall and precision metrics, we compared all the human labeling and machine labeling with the ground truth, which was generated by two experienced researchers in loop labeling (see the “Data preparation and augmentation” section). Note that these two researchers were not in the group of five researchers who generated the human comparison; thus these two researchers were not used in assessing recall and precision vs. the ground truth, as their assessments were used to determine the ground truth.

Figure 3 shows the comparison of defect size distribution metrics analyzed by five researchers and by our automated program. Figure 4 shows the defects recognized by the

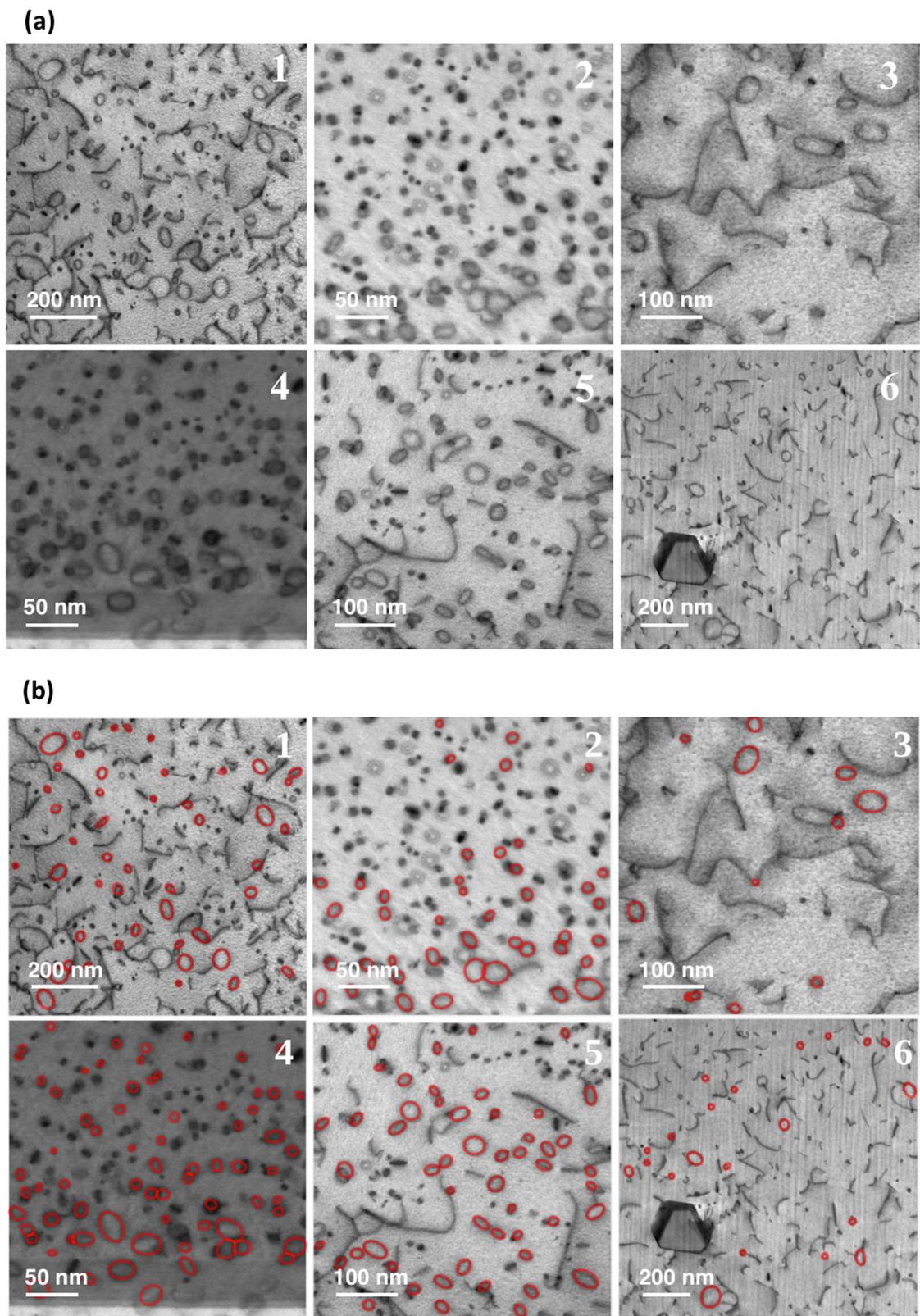
automated program on the six images selected from among the test images and used for the human/machine comparison. We can see that overall the machine did an excellent job of determining a total loop number similar to that provided by human labeling. For mean diameter, the automated defect detection algorithm also showed excellent agreement with the human labeling, except perhaps in images 3 and 4. The automated defect detection algorithm obtained a slightly lower value in image 3, as it made an error and missed a very large loop with a high aspect ratio.

The reason that the program missed the large loop could be inadequate training instances of large loops with high aspect ratios, as this type of defect is uncommon in the provided micrographic images. While more image augmentation targeted at larger loops could potentially reduce this error, we believe further optimization is better left to a future study that also explores additional methods, multiple defects, and larger and more varied input data sets (e.g., from multiple instruments and materials). The algorithm obtained a slightly lower value in image 4 because the machine tends to give more loop predictions for this type of dark, blurred image, and usually these loops have smaller diameters than the average.

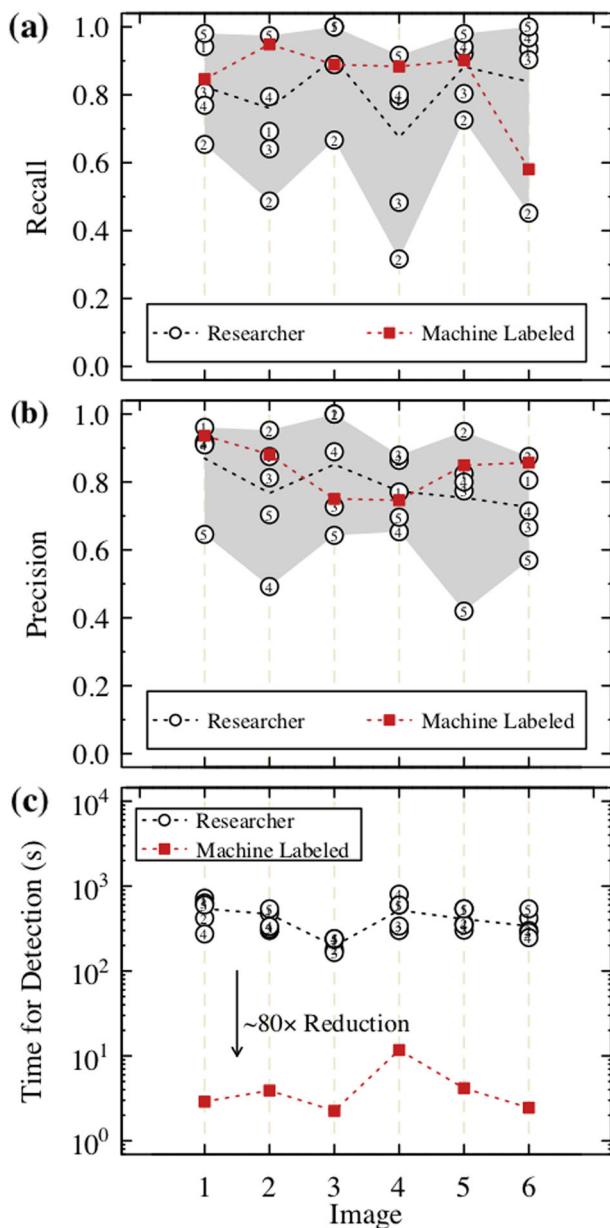
To obtain more detailed understanding in how errors in predictions correlate with test image characteristics, we evaluated the automated model recall and precision score for loop defects with varying brightness, contrast, and loop size. The analysis revealed that there was no direct correlation of model performance with the variables. The model performance was rather highly influenced by the frequency of loops in the data set, i.e., the more frequent a certain type of loops appears (e.g. loops of typical brightness) in the data set, the better the automated model performs for that type. Figures with recall and precision scores as a function of different image characteristics for the test images are provided in the Supplemental Information (SI) Section I. Another possible source of error is the overlapping loops. The automated model gets correct identification for small overlaps or closely touching loops that we have in the test data set (see some loops in Fig. 4b(2) and Fig. 4b(4)). However, we can also see that the automated model can experience systematic issues with large overlaps of loop defects. Large overlapped loops are rare in our data set but we do find that our model was only able to identify one of the large overlapped loops in some examples we considered (See one example in Fig. 4b(5)).

The machine also was in very good agreement with human labeling for standard deviation, again except perhaps for image 3. The error in the standard deviation for image 3 determined using the algorithm can be explained in the same way as the observed issue in diameter for image 4—there were few large, outlier loop sizes. Although these discrepancies suggest ways to improve the machine labeling, the predictions from the model were approximately within the range for the distribution of the five human predictions for every case except the loop diameter for image 3. These results demonstrate the capability of the model to obtain reasonable results regarding loop shape properties over a wide range of images.

Figure 5 compares the recall, precision, and time efficiency of the labeling from the five researchers and the machine compared with the ground truth labeling (see “Data preparation and augmentation” section). The average recall of the five researchers over six images was  $0.804 (\pm 0.029)$  (parentheses represent one standard deviation from the mean), and the average precision of researchers was  $0.790 (\pm 0.023)$ . The average recall of the machine was  $0.842 (\pm 0.054)$  and the average precision of the machine was  $0.837 (\pm 0.031)$ . These larger values for recall and precision for machine vs. human demonstrate that the machine had higher performance than the human average on the six-image test set. The success of the machine performance vs. human performance clearly demonstrated the high quality of the machine model vs. the ground truth, and it also demonstrated the limitations of



**Fig. 4** **a** Images selected from the complete test set. The selected images were also analyzed by five researchers for the comparison with machine labeling. **b** Fitted ellipse labeled by the automated machine learning program on the six selected test images. Image size: 1024 × 1024 pixels for all images except image 4, 2048 × 2048 pixels for image 4



**Fig. 5** Comparison of **a** recall, **b** precision, and **c** time efficiency of both the human and machine labeling. The image number corresponds to the image number in Fig. 4

human assessment. Specifically, different researchers have large discrepancies in defect labeling. A major cause of this discrepancy is likely that researchers differ in how they analyze ambiguous loops and given researchers might have differed in the way they treated ambiguous loops in the ground truth labeling. Based on Fig. 5c, the average time for detection for the researchers is 2440 ( $\pm 555$ ) s, and the average time for detection for the ML approach is 27 ( $\pm 1$ ) s,  $>80$  times faster than human labeling (values in parentheses represent the standard deviations of the times). Machine labeling can be further accelerated by allowing more central processing unit (CPU) and graphics processing unit (GPU) cores to compute in parallel. The comparisons in Figs. 3 and 5 prove that automated image analysis can perform comparably to the human average. The comparison also indicates that the automated machine tool provides more consistent analysis results—reliable, reproducible results across different sets of micrographs, microscopes, and research facilities—compared with the

large human variability in labeling test images for which the microscopic technique and research facility were held constant. Furthermore, while this test is not definitive regarding the relative accuracy of the automated vs. manual labeling, it does show how the ML approach can be trained to represent the skills of the most experienced researchers. With careful training, this characteristic can be a significant advantage over manual labeling, especially for beginners within the field.

## DISCUSSION

This study demonstrated and evaluated an automated approach to dislocation loop defect detection using contemporary ML, computer vision, and image analysis techniques. The developed approach achieved similar performance to the human average across the same data set. Our results indicate that computer vision techniques are very promising for microscopic image analysis to replace human labor and produce standard analysis output. Developing automated image analysis techniques could greatly reduce human labor and time in labeling images and reduce variability compared with human labeling, both of which would benefit the microscopic community.

The current model was trained with only a limited number of micrographic images, and the performance could be further improved by supplying more well-annotated images. We can also envision that the automated model can become more robust to systematic issues when trained with more micrographs of various brightness, contrast, focus, and defect size scales produced by different researchers for a range of materials, instruments, and imaging modes. We note that the model has only been trained on one material and evaluated on the same material. For materials with similar looking micrographs, we would expect good performance of the model. But for materials that have defects that look similar to loops in some way, e.g., extensive background dislocations, or that have very different environments (e.g., many overlapping black dots or precipitates on the loops), the model might give very poor results. However, we expect that, as models like this are further developed and exposed to orders of magnitude more data, transferability to new materials, microscopes, irradiation conditions, etc., will be very robust.

The current model was developed using only a single CPU and GPU. Increased CPU/GPU performance and parallel image processing could further accelerate the speed. Speeds could potentially be increased sufficiently to allow real-time image recognition, which could be embedded in the electron microscopic system and provide in situ analysis directly on the monitor screen. Such a system would enable researchers to adjust their characterizations in real time in response to the data. Although our approach was primarily built for loop defect recognition, creating new models to detect other types of defects or patterns in micrographic images is worth exploring if micrographic images containing those types of defects can be obtained experimentally or simulated with high fidelity. We propose that an online system in which any researcher can contribute annotated images for training the model could be an effective way to develop such models. We envision that automated image recognition can dramatically change the current microscopic characterization workflow by allowing orders of magnitude more images to be fully analyzed automatically and nearly instantly.

## METHODS

### Data set collection

Data set collection was completed as part of a large-scale effort to characterize iron–chromium–aluminum (FeCrAl) materials neutron-irradiated within the High Flux Isotope Reactor at Oak Ridge National Laboratory. The data set comprises a series of published<sup>40–43</sup> and unpublished data. Collection was completed over 3 years and spans a

range of different FeCrAl alloys, including model, commercial, and engineering-grade alloys irradiated to light water reactor-relevant conditions (e.g., <15 displacements per atom and temperatures of nominally 285–320 °C).

All images were generated from focused ion beam-prepared samples with thickness in the range of 40–175 nm and imaged using a JEOL JEM-2100F field emission gun STEM (FEG-STEM) operating at 200 kV. Similar imaging has been completed using other FEG-STEM instruments but was not included in the database. Various techniques can image the elliptical loops that were of interest in this study, including two-beam and weak-beam dark-field imaging,<sup>1,44</sup> but these more conventional techniques are prone to background variation due to elastic contrast. To quickly and rapidly provide images with thousands of dislocation loops over large areas (>1–5 microns) over a wide range of magnifications (100–500kx), the on-zone STEM imaging technique was applied.<sup>35,45</sup> All images within the data set were collected by imaging down the [100] zone axis, which allowed radiation-induced defects with a Burgers vector of  $a/2\langle 111 \rangle$  to be imaged as ellipses in the two-dimensional projection of the volume.<sup>46</sup> All images were taken using the bright-field detector, resulting in defects appearing as black foreground on a white background. Images were taken with varying magnification, detector resolution, and pixel dwell time to limit instrument biasing into the data set. Negative images were taken in the same manner on unirradiated reference samples.

### Data set preparation

For the data set used to train the cascade object detector, we used an open-source software called ImageJ<sup>39</sup> to manually annotate the positions of loop defects on the microscopic images. We generated labeled files containing the bounding boxes of loops for each corresponding image. For the data set used to train the CNN, we applied the cascade object detector to the micrograph images in the augmented training set and compared the detector predictions with ground truth labeling. Each micrograph image had 10–200 loops inside. We grouped the correct predictions as a set of cropped images with loops and grouped the false predictions as a set of cropped images without loops. The CNN training set had 60,000 images of  $64 \times 64$  pixels in total, among which 30,000 images contained loops and 30,000 images contained no loops. Typically, a CNN can take images with red, blue, and green channels through an input layer of width by height by number of channels. Here, as our microscopic images was gray-scale (only one channel of intensity), we provided more information to the CNN by adding two more layers to each image—an increased-contrast image with 8% pixel intensity saturated to maximum/minimum and a Gaussian blurred image. Thus the size of each image was  $64 \times 64 \times 3$ . The purpose of adding two more layers in each image was to provide more information regarding various contrast levels or blurring.

### Model training

The cascade object detector used integral image representation for fast feature extraction<sup>16</sup> from original images, and it trained simple and efficient classifiers using the adaptive boosting (AdaBoost) method<sup>8</sup> to select a small number of important features. Successively, more complex classifiers were then combined in a cascade structure, which dramatically increased the detection speed by focusing attention on promising regions of the image. For the feature extraction method, the LBP<sup>47</sup> was used as the feature type to encode local edge information into feature vectors. The LBP feature was selected as a compromise between training time and detector performance, compared with the slower Haar<sup>16</sup> and faster and less accurate HOG<sup>6</sup> feature extraction methods. We trained a 40-stage cascade object detector with the augmented training set and negative images. We set the maximum false positive rate to 0.3 and the minimum true positive rate to 0.997 to make the overall true positive prediction as high as possible and obtain fair precision performance. We used the cascade object detector training function provided by Matlab computer vision toolbox.<sup>48</sup> The detector training time grew with the number of images in the data set. With the augmented training data set, we trained the 40-stage cascade object detector on a 6-core Intel Xeon E5-1650 CPU for about 6 days.

The CNN was trained within the deep learning package from Matlab. We trained the network on a single GTX 1070 GPU. We constructed our own neural network structure based on Matlab's deep learning tutorial.<sup>49</sup> We tuned our network structure by varying the number of layers, size, and stride of filters in each layer, the number of maxpool layers, and the size of fully connected layers to find the best model for micrographic screening

purpose. We determined the final CNN structure by the performance on the validation set. We constructed our CNN as one input, three convolution, four nonlinear, three maximum pooling, two fully connected layers, one softmax layer, and one class-output layer. The size of the image input layer was  $64 \times 64 \times 3$  with zero center normalization, followed by convolution layers containing 64 filters with size of  $10 \times 10 \times 3$ , stride [1 1], and padding [2 2]. We used the ReLu (rectified linear unit) function to provide nonlinearity after convolution layer, and we applied maximum pooling with a filter size of  $3 \times 3$  and a stride of [2 2] and padding [0 0] after ReLu layer. We repeated the block of convolution, ReLu, and maximum polling layers twice. The filter size, stride, and padding for each repeated layer were the same as the corresponding layer in the first block, except the next convolution layer contained 64 filters with size of  $10 \times 10 \times 64$  and the last convolution layer contained 128 filters with the size of  $10 \times 10 \times 64$ . After three blocks of convolution-ReLu-MaxPool layers, we used a fully connected layer of 64 nodes followed by a ReLu layer and another fully connected layer of 2 nodes as the classification layer with the classes of loop and non-loop. For the final output node, we used the softmax function to give the output category for each input image. We trained the CNN on the CNN training set generated using the 40-stage cascade object detector. We started training the CNN initialized with randomly assigned weights, and data were fed into the CNN in a mini-batch size of 128. The cross-entropy loss for binary classification was optimized with Stochastic Gradient Descent with Momentum as the optimizer, with the momentum set to 0.9 and L2 Regularization set to 0.08. The initial learning rate is 0.001 with the learning rate drop factor of 0.05 and learning rate drop period is 8. Considering human mistakes in the training set labeling, we set the training accuracy stop threshold to 0.95 to avoid over-fitting. The total training finished in 5 h on a Nvidia GTX1070 GPU.

### Data availability

The data of training images and the source code that support the findings of this study are available in Materials Data Facility with the identifier doi:10.18126/M2692Z.<sup>50</sup> The source code is also available on Github (<https://github.com/uw-cmg/MATLAB-loop-detection>).

### ACKNOWLEDGEMENTS

We would like to acknowledge researchers who shared negative images and offered help for labeling the test images (in alphabetical order by last name): Hima Bharathi Adusumilli, Sam Briggs, Tianyi Chen, Boopathy Kombaiah, Stephen Taller, Chris Ulmer, Xing Wang, and Dalong Zhang. Research was sponsored by the Department of Energy (DOE) Office of Nuclear Energy, Advanced Fuel Campaign of the Nuclear Technology Research and Development program (formerly the Fuel Cycle R&D program). Neutron irradiation of FeCrAl alloys at Oak Ridge National Laboratory's High Flux Isotope Reactor user facility was sponsored by the Scientific User Facilities Division, Office of Basic Energy Sciences, DOE. Support for D.M. was provided by the National Science Foundation Software Infrastructure of Sustained Innovation (SI2) award No. 1148011.

### AUTHOR CONTRIBUTIONS

Initial project ideas were from K.F. and D.M. Image collection was performed by K.F. Data set labeling was performed by K.F. and W.L. Coding and model training were performed by W.L. Human survey design, interpretation of the result, and manuscript writing were contributed together by W.L., D.M., and K.F.

### ADDITIONAL INFORMATION

**Supplementary information** accompanies the paper on the *npj Computational Materials* website (<https://doi.org/10.1038/s41524-018-0093-8>).

**Competing interests:** The authors declare no competing interests.

**Publisher's note:** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

### REFERENCES

1. Jenkins, M. L. & Kirk, M. A. *Characterisation of Radiation Damage by Transmission Electron Microscopy*. (Boca Raton, FL, CRC Press, 2000).
2. Zinkle, S. J. & Busby, J. T. Structural materials for fission & fusion energy. *Mater. Today* **12**, 12–19 (2009).

3. Jesse, S. et al. Big data analytics for scanning transmission electron microscopy ptychography. *Sci. Rep.* **6**, 26348 (2016).
4. Lowe, D. G. Object recognition from local scale-invariant features. In *Proc. of the Seventh IEEE International Conference on Computer Vision*. 1150–1157 (IEEE, Kerkyra, Greece, 1999).
5. Zhao, J., Kong, Q.-J., Zhao, X., Liu, J. & Liu, Y. A method for detection and classification of glass defects in low resolution images. In *2011 Sixth International Conference on Image and Graphics (ICIG)*. 642–647 (IEEE Computer Society Washington, DC, USA 2011).
6. Dalal, N. & Triggs, B. Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 886–893 (IEEE, San Diego, CA, USA 2005).
7. Yu, K., Jia, L., Chen, Y. & Xu, W. Deep learning: yesterday, today, and tomorrow. *J. Comput. Res. Dev.* **50**, 1799–1804 (2013).
8. Viola, P. & Jones, M. Rapid object detection using a boosted cascade of simple features. In *Proc. of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1, 511–518 (IEEE, Kauai, HI, USA 2001).
9. Belongie, S., Malik, J. & Puzicha, J. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**, 509–522 (2002).
10. Felzenszwalb, P. F., Girshick, R. B., McAllester, D. & Ramanan, D. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**, 1627–1645 (2010).
11. Ren, S., He, K., Girshick, R. & Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, *Advances in Neural Information Processing Systems*. (eds Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M. & Garnett, R.) In *Neural Information Processing Systems 2015*. 91–99 (NIPS, Montréal, Québec, Canada, 2015).
12. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
13. Viola, P. & Jones, M. J. Robust real-time face detection. *Int. J. Comput. Vision* **57**, 137–154 (2004).
14. LeCun, Y. et al. Handwritten digit recognition with a back-propagation network. (ed Touretzky, D.S.) In *Advances in Neural Information Processing Systems*. 396–404 (NIPS, Denver, Colorado, United States 1989).
15. Yang, Z., Tao, D.-p., Zhang, S.-y. & Jin, L.-w. Similar handwritten Chinese character recognition based on deep neural networks with big data. *J. Commun.* **35**, 184–189 (2014).
16. Sharifara, A., Rahim, M. S. M. & Anisi, Y. A general review of human face detection including a study of neural networks and Haar feature-based cascade classifier in face detection. In *2014 International Symposium on Biometrics and Security Technologies (ISBAST)*. 73–78 (IEEE, Kuala Lumpur, Malaysia 2014).
17. Jiang, H. & Learned-Miller, E. Face detection with the faster R-CNN. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*. 650–657 (IEEE, Washington, DC, USA 2017).
18. Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning*. (MIT Press, Cambridge, Massachusetts, USA 2016).
19. Zeiler, M. D. & Fergus, R. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*. 818–833 (Springer, Cham, Zurich, Switzerland 2014).
20. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. Preprint at <http://arXiv:1409.1556> (2014).
21. Szegedy, C. et al. Rethinking the Inception architecture for computer vision. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. 2818–2826 (IEEE, Las Vegas, NV, USA 2016).
22. Long, J., Shelhamer, E. & Darrell, T. Fully convolutional networks for semantic segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. 3431–3440 (IEEE, Boston, MA, USA, 2015).
23. He, K., Gkioxari, G., Dollár, P. & Girshick, R. Mask r-cnn. Preprint at <http://arXiv:1703.06870> (2017).
24. Kreshuk, A. et al. Automated detection and segmentation of synaptic contacts in nearly isotropic serial electron microscopy images. *PLoS ONE* **6**, e24899 (2011).
25. Ciresan, D., Giusti, A., Gambardella, L. M. & Schmidhuber, J. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in Neural Information Processing Systems*. 2843–2851 (NIPS, Stateline, Nevada, United States 2012).
26. Ciresan, D. C., Giusti, A., Gambardella, L. M. & Schmidhuber, J. Mitosis detection in breast cancer histology images with deep neural networks. In *International Conference on Medical Image Computing and Computer-assisted Intervention*. 411–418 (Springer, Berlin, Heidelberg, Nagoya, Japan 2013).
27. Lubbers, N., Lookman, T. & Barros, K. Inferring low-dimensional microstructure representations using convolutional neural networks. *Phys. Rev. E* **96**, 052111 (2017).
28. Belianinov, A. et al. Big data and deep data in scanning and electron microscopies: deriving functionality from multidimensional data sets. *Adv. Struct. Chem. Imaging* **1**, 6 (2015).
29. DeCost, B. L., Francis, T. & Holm, E. A. Exploring the microstructure manifold: image texture representations applied to ultrahigh carbon steel microstructures. *Acta Mater.* **133**, 30–40 (2017).
30. Chowdhury, A., Kautz, E., Yener, B. & Lewis, D. Image driven machine learning methods for microstructure recognition. *Comput. Mater. Sci.* **123**, 176–187 (2016).
31. Ziatdinov, M. et al. Deep learning of atomically resolved scanning transmission electron microscopy images: chemical identification and tracking local transformations. *ACS Nano* **11**, 12742–12752 (2017).
32. Xu, W. & LeBeau, J. M. A deep convolutional neural network to analyze position averaged convergent beam electron diffraction patterns. Preprint at <http://arXiv:1708.00855> (2017).
33. Vincent, L. & Soille, P. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. In *IEEE Transactions on Pattern Analysis & Machine Intelligence* 583–598 (IEEE, 1991).
34. Mathworks. *regionprops: Measure Properties of Image Regions* <https://www.mathworks.com/help/images/ref/regionprops.html> (2017).
35. Parish, C. M., Field, K. G., Certain, A. G. & Wharry, J. P. Application of STEM characterization for investigating radiation effects in BCC Fe-based alloys. *J. Mater. Res.* **30**, 1275–1289 (2015).
36. Turcot, P. & Lowe, D. G. Better matching with fewer features: The selection of useful features in large database recognition problems. In *2009 IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops)*. 2109–2116 (IEEE, Kyoto, Japan 2009).
37. Murphy, K. P. *Machine Learning: A Probabilistic Perspective*. (MIT Press, Cambridge, MA 2012).
38. Gonzalez, R. C. & Woods, R. E. *Digital image analysis*. 3rd edition (Prentice-Hall, Upper Saddle River, New Jersey, 2008).
39. Abràmoff, M. D., Magalhães, P. J. & Ram, S. J. Image processing with ImageJ. *Biophotonics Int.* **11**, 36–42 (2004).
40. Field, K. G., Hu, X., Littrell, K. C., Yamamoto, Y. & Snead, L. L. Radiation tolerance of neutron-irradiated model Fe–Cr–Al alloys. *J. Nucl. Mater.* **465**, 746–755 (2015).
41. Field, K. G. et al. Heterogeneous dislocation loop formation near grain boundaries in a neutron-irradiated commercial FeCrAl alloy. *J. Nucl. Mater.* **483**, 54–61 (2017).
42. Field, K. G., Briggs, S. A., Sridharan, K., Yamamoto, Y. & Howard, R. H. Dislocation loop formation in model FeCrAl alloys after neutron irradiation below 1 dpa. *J. Nucl. Mater.* **495**, 20–26 (2017).
43. Briggs, S. A., Sridharan, K. & Field, K. G. Correlative microscopy of neutron-irradiated materials. *Adv. Mater. Process.* 174(10) (2016).
44. Kirk, M., Yi, X. & Jenkins, M. Characterization of irradiation defect structures and densities by transmission electron microscopy. *J. Mater. Res.* **30**, 1195–1201 (2015).
45. Phillips, P., Brandes, M., Mills, M. & De Graef, M. Diffraction contrast STEM of dislocations: imaging and simulations. *Ultramicroscopy* **111**, 1483–1487 (2011).
46. Yao, B., Edwards, D. J. & Kurtz, R. J. TEM characterization of dislocation loops in irradiated bcc Fe-based steels. *J. Nucl. Mater.* **434**, 402–410 (2013).
47. Ahonen, T., Hadid, A. & Pietikäinen, M. Face description with local binary patterns: Application to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**, 2037–2041 (2006).
48. Alionte, E. & Lazar, C. A practical implementation of face detection by using Matlab cascade object detector. In *2015 19th International Conference on System Theory, Control and Computing (ICSTCC)*. 785–790 (IEEE, Cheile Gradistei, Romania 2015).
49. Mathworks. *Object Detection Using Deep Learning* <https://www.mathworks.com/help/vision/examples/object-detection-using-deep-learning.html> (2017).
50. Li, W., Field, K. G. & Morgan, D. *Detection of Open Loop Defects in STEM Images of Irradiation-Damaged Alloys – Source Code for Detection and Image Dataset* (Materials Data Facility, doi: <https://dx.doi.org/doi:10.18126/M2692Z>, 2018).



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.