Long *et al. Heritage Science* (2022) 10:150
https://doi.org/10.1186/s40494-022-00778-3

Heritage Science

RESEARCH Open Access

Automated corrosion detection in Oddy test coupons using convolutional neural networks

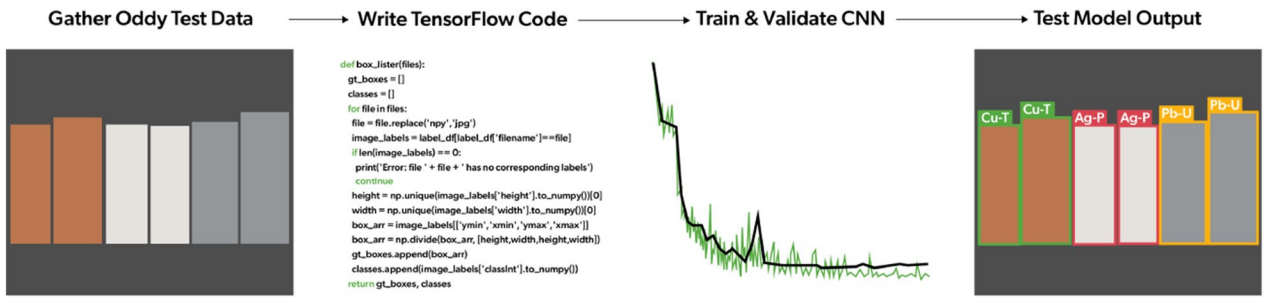
Emily R. Long[1,2*], Alayna Bone[3], Eric M. Breitung[3], David Thickett[4] and Josep Grau-Bové[1]

Abstract

The Oddy test is an accelerated ageing test used to determine whether a material is appropriate for the storage, transport, or display of museum objects. The levels of corrosion seen on coupons of silver, copper, and lead indicate the material's safety for use. Although the Oddy test is conducted in heritage institutions around the world, it is often critiqued for a lack of repeatability. Determining the level of corrosion is a manual and subjective process, in which outcomes are affected by differences in individuals' perceptions and practices. This paper proposes that a more objective evaluation can be obtained by utilising a convolutional neural network (CNN) to locate the metal coupons and classify their corrosion levels. Images provided by the Metropolitan Museum of Art (the Met) were labelled for object detection and used to train a CNN. The CNN correctly identified the metal type and corrosion level of 98% of the coupons in a test set of the Met's images. Images were also collected from the American Institute for Conservation's Oddy test wiki page. These images suffered from low image quality and were missing the classification information needed to train the CNN. Experts from cultural heritage institutions evaluated the coupons in the images, but there was a high level of disagreement between expert classifications. Therefore, these images were not used to train the CNN. However, the images proved useful in testing the limitations of the CNN trained on the Met's data when applied to images of coupons from different Oddy test protocols and photo documentation procedures. This paper presents the effectiveness of the CNN trained on the Met's data to classify Met and non-Met Oddy test coupons. Finally, this paper proposes the next steps needed to produce a universal CNN-based classification tool.

Keywords: Object detection, Machine learning, Computer vision, Preventive conservation

Graphic Abstract



*Correspondence: emily.long.17@alumni.ucl.ac.uk

[1] Institute for Sustainable Heritage, University College London, London, UK
Full list of author information is available at the end of the article

Springer Open

© The Author(s) 2022. **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 licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence 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 licence, visit http://creativecommons.org/licenses/by/4.0/. The Creative Commons Public Domain Dedication waiver (http://creativecommons.org/publicdomain/zero/1.0/) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

Introduction

The Oddy test is an accelerated ageing test used to determine whether a material is safe for use in the storage, transport, or display of museum objects [1]. In the original test, a small sample of a test material, such as fabric or tape, is placed at the bottom of a reaction vessel along with a small amount of water. A rectangular coupon of metal, either silver (Ag), copper (Cu), or lead (Pb), is hung above the material. The sealed vessel is heated at 60° C for 28 days to accelerate the emission of reactive chemicals that may be created in a museum setting over time. The level of corrosion on the metal coupons indicates the corrosivity of the emissions from the material and thus the material's safety for use near museum objects [2, 3]. Despite the Oddy test's widespread use, it is often critiqued for a lack of standardisation, identification of corrosive emission types, and objectivity. This paper proposes a method of improving the objectivity of the last stage of the Oddy test, the visual evaluation, based on training a convolutional neural network (CNN) to process photographic images of the coupons.

The Oddy test evolved from the original version developed by W.A. Oddy in 1973 [1]. Instead of only one metal coupon per test, the Metropolitan Museum of Art (the Met) [4] and the British Museum (BM) [5] both developed '3-in-1' versions where one reaction vessel contains all three silver, copper and lead coupons. The original glass stoppers [4] were replaced by silicone stoppers [5] or 3D printed nylon hangers [2]. While the most recent protocols from the BM and the Met are described thoroughly and address many of the factors that lead to inconsistent results for Oddy testing, over 20 other test methods are used in museums throughout the world [2, 3, 6, 7]. Some Oddy test procedures vary significantly from the original protocol with, for example, coupons being in contact with the material or removed from the reaction vessel before they have reached the full 28 days [8]. In an experiment conducted by the BM, researchers found that for the same test material, different Oddy test procedures can lead to different levels of corrosion [9]. This makes it difficult to compare tests between institutions. Some attempts to standardise the Oddy test have involved training sessions, publicly accessible test results and procedures [10, 11], and plans to develop bundles of equipment for purchase [7]. However, the Oddy test procedure remains unstandardised.

Overall, the Oddy test is an effective and conservative assessment that prevents unsuitable materials from being used to display or store heritage objects [12]. The British Museum, for instance, claims that there were only two occasions over twenty years where silver corrosion occurred on displayed objects when the Oddy test was used [3]. However, the Oddy test does not identify the corrosive compounds off-gassed by a test material. Researchers have used gas chromatography-mass spectrometry (GC-MS) to test for reactive compounds after museum objects have been damaged during display. For example, some adhesives that had passed the Oddy Test were found to produce a volatile compound called tetramethylpiperidinol (TMP-ol) which reacted with acids in the case to form crystalline deposits within display cases at the Museum of Fine Arts, Boston [13, 14], the Smithsonian's National Museum of the American Indian [15], and the Rijksmuseum [16]. Similarly, corrosion of copper and silver objects in the Indianapolis Museum of Art (IMA) was linked to PVC boards. GC-MS of the installed board was used to establish that ethylhexyl thioglycolate caused the corrosion. [17]. To vet materials using GC-MS, scientists currently require 'chemical intuition' to connect the identified chemicals to corrosive effects [18]. The Oddy test can help build this knowledge by identifying test materials that produce different types of corrosion on each metal [19]. The Oddy test can also be used to establish which chemicals affect each metal at particular concentrations [20]. Thus, the Oddy test will still be used alongside GC-MS as a tool for selecting materials that are safe for use with collections.

The Oddy test is also critiqued for the lack of objectivity when testers evaluate the corrosion levels of the coupons. While recent research suggests that oxygen depletion testing within the Oddy test could be an alternative and quantitative method for evaluating materials [21], widely accessible methods for reducing the subjectivity of Oddy test coupons are still needed. To evaluate Oddy test coupons, operators perceive colour and texture differently and incorporate individual bias into their evaluations. The Met has provided annotated libraries of coupon images for reference during coupon evaluation [22]. These libraries contain 82 images, but the highlighted corrosion types and categorizations are specific to the Met's test protocol. For a universal system, coupon images and classifications with other types of corrosion are required [9]. Since the use of visual perception to interpret reference libraries is a subjective process, alternatives are needed to improve the objectivity of metal coupon classification. Automated computer-based classification systems are capable of utilizing reference images to train a neural network to classify corrosion levels. In previous research, the Material Checker (MATCH) project proposed combining photography hardware with an artificial neural network to identify coupon corrosion levels [7]. However, this hardware would require museums to change their method of documenting the coupons. Alternatively, this paper will consider a type of neural network that could be more easily integrated into any institution's workflow.

When given an input of Oddy test images, convolutional neural networks for object detection can be trained to identify individual coupons and classify different levels of corrosion. CNNs are capable of learning patterns even in very abstract images, such as close-up images of steel and copper [23, 24]. CNNs have also successfully identified corrosion in grounding grids [25], civil infrastructure [26, 27], the hulls of ships [28], and gas pipelines [29]. Object detection CNNs have been useful within both metal corrosion and heritage contexts. For built heritage, object detection CNNs have identified defects in brick masonry [30], missing pieces from ancient roofs [31], and architectural elements like arches and columns [32]. Given their effectiveness in other applications, CNNs are a promising addition to the Oddy test workflow.

The goal of this study is to determine how an object detection CNN can be used to reduce the subjectivity in the interpretation of Oddy tests. This paper summarises the first phase of research, which aims to:

1 Gather image data from multiple sources and prepare it for the application of object detection;
2 Train an object detection CNN and evaluate its performance for the detection and differentiation of corrosion levels in Oddy tests;
3 Establish directions for future research in terms of data and CNN requirements.

The long-term aim is to create a publicly accessible Oddy coupon assessment tool. On top of improving the objectivity of the Oddy test, this tool should be designed to be time-saving for users. The tool should also be instructional, so that it is a suitable alternative for coupon image reference libraries. To achieve this aim, this paper will train an object detection CNN on Oddy test image data and establish actions for further research.

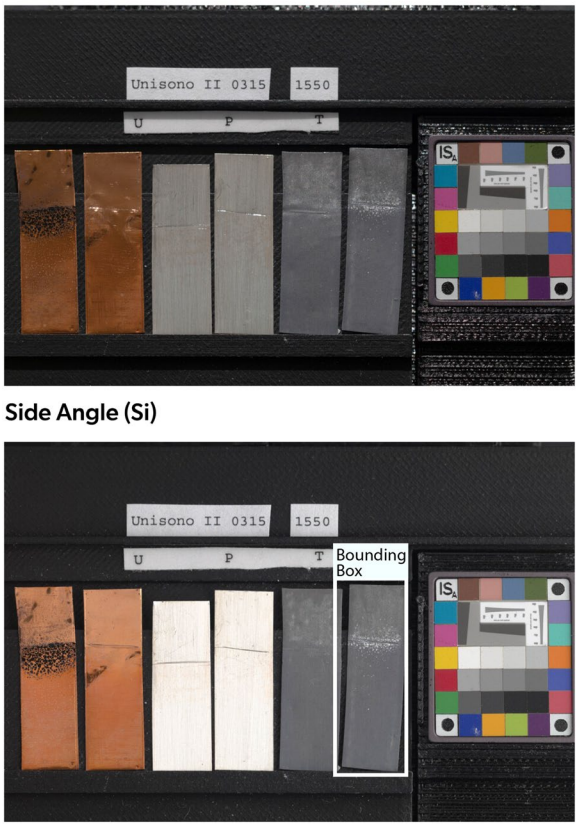


Fig. 1 The Met's glancing and side angle lighting on Oddy test coupons for sample number 1550. The Si angle includes an example of a bounding box label for the CNN

Methods

Data

Images from the Met

The Met provided 2208 images for this study. The Met has a sophisticated protocol for photographing the coupons [22]. For each test, the coupons are placed under the two lighting conditions shown in Fig. 1. Glancing angle (GA) lighting highlights differences in surface texture, while the side angle (Si) lights coupons more evenly [22]. Variations in image lighting are helpful for training a CNN, so both types of images were used.

In the images, coupons are placed next to printed ratings of their corrosion levels. Permanent (P) indicates that a test material can be used near collections indefinitely. Temporary (T) means that a material can be used for up to six months. Unsuitable (U) indicates that a material should not be used for the storage or display of museum objects [22]. The Met also summarised these ratings in a spreadsheet alongside comments for each coupon, such as in Table 1. Most of the corrosion levels were documented when the photos were taken, but over 350 images, particularly control tests, were not originally assigned ratings within the image or spreadsheet. For this research, the images without ratings were evaluated, and their coupon corrosion levels were added to the spreadsheet.

Images from the AIC wiki page

The American Institute for Conservation (AIC) hosts a wiki page with a table where institutions can share results from their Oddy tests [33]. This table was downloaded by converting it from HTML to a CSV file. Of the 2561 rows in the table, only 616 rows linked to images. The majority of those images are from the Met, so they were not downloaded from the wiki to avoid duplicates. In total, 170 images were downloaded from the Autry Museum, Cleveland Museum of Art (CMA), Hodgkins et al. (HOD)

[34], Heritage Conservation Centre Singapore (HCC), and the New York University Libraries (NYUL) [33].

The wiki table gives an overall rating of permanent, temporary, or unsuitable for a test. However, it does not provide coupon-specific ratings such as those shown in Fig. 1, which were provided directly from the Met and can be found in Met images on the Wiki. To train an object detection CNN, each coupon required a rating. Therefore, the wiki images were reviewed by a group of experts from heritage institutions. Experts were only provided with the images, the information from the wiki table, the overall image classification, and a spreadsheet for the ratings. Using the programming language R, their ratings were combined by identifying the most frequently suggested expert rating per coupon. However, some images had inconclusive results with multiple potential labels per coupon. See the "Results from AIC wiki data" section for the results on this dataset.

Bounding box labels

Classification CNNs assign a single class to an image that describes its subject. Object detection CNNs detect multiple objects within an image and assign a class to each [35]. The Oddy test CNN will be trained to identify nine classes based on the metal types and corrosion ratings: Ag-P, Ag-T, Ag-U, Cu-P, Cu-T, Cu-U, Pb-P, Pb-T, and Pb-U.

Object detection CNNs require images to be labelled with ground truth bounding boxes around the objects. A bounding box example is shown in Fig. 1. Bounding boxes for the coupons were drawn onto the Oddy test images using the tool MakeSense [36]. This is a manual, time-consuming task. The labels were exported as a CSV file. Examples of the bounding box labels are shown in Table 2. The bounding box coordinates (xmin, ymin) and (xmax, ymax) represent the top left and bottom right points of the coupon. MakeSense outputs the coordinates (xmin, ymin), width, and height of the bounding boxes. These quantities can be added in R or Python to get the maximum x and y values.

Convolutional neural networks

A convolutional neural network is a type of machine learning model that learns patterns from images. First, images are represented by matrices of pixel colour values. In a CNN layer, matrices with values called weights are multiplied against the image matrix, and then the calculated values are passed on to the next layer [35]. When a CNN is trained, these weights are adjusted to extract different kinds of features from the images based on the desired output [37]. Since CNN architectures are complex but well documented [35, 37], this paper will focus on their implementation with Python and the results on the Oddy test data.

Convolutional neural networks require significant time and computational power to train. Running Python scripts to train a CNN on a local computer could take days or weeks. One solution is to use Google Colab, a cloud-based service that runs Python notebooks within an internet browser [38]. Colab gives free, albeit limited, access to graphical processing units (GPUs) which can train CNNs within a few hours. This study utilised Google Colab Pro which is an affordable monthly service that offers higher RAM memory and fewer restrictions on GPU use.

Within Python, the package Tensorflow has many features that enable efficient machine learning projects. The TensorFlow Object Detection API page on GitHub contains essential tutorials and Python scripts [39]. TensorFlow also has a library of object detection models, called the model zoo [40], that have been trained with the COCO dataset [41]. Since the Met's dataset is very small compared to the 200,000 images in the COCO dataset, transfer learning is essential. Transfer learning is a method of using pre-trained weights downloaded from Tensorflow as a starting point for training the model on smaller datasets [40]. The Python notebooks used for training and testing TensorFlow models are available in this paper's GitHub repository [42].

Image preparation in Python

The images were randomly split using R into three sets with different purposes. The largest set, the training set, was used to update the weights within the CNN. Validation set images were used to the track a CNN's progress and select suitable models, but they were not used to update the weights. Finally, the test set was used only to test the final model [43].

CNN architectures start with images of a particular size. For example, the TensorFlow object detection models are developed for specific image sizes ranging from 320×320 to 1536×1536 pixels [40]. Models with larger input images tend to be more precise, but models with smaller images are more computationally efficient [44]. TensorFlow models will automatically resize any inputted images to the required size for the model architecture.

One method for using image data in CNNs is to first load JPG files into Python, then convert them to NumPy arrays, and finally convert them to tensors. NumPy arrays are a type of array from the Python package NumPy that can represent multidimensional image data. Tensors are also multidimensional arrays, similar to NumPy arrays, but are formatted for TensorFlow models [45].

The Met's images were initially up to 6000×4000 pixels. Due to the large number of pixels, these images were

slow to load into Python (30 seconds per image). The images were resized to a width of 1536 pixels, the maximum size for TensorFlow's object detection models [40]. This drastically sped up the loading process (2.4 seconds per image), but for this size, Google Colab would run out of memory when a large set of NumPy arrays were converted to tensors. To save on memory, the images were further reduced to a height of 640 pixels, which is another common model image size [40]. It still took just under an hour to load the 2208 JPG files at 960 × 640 pixels into Python. This would be an inefficient step to repeat every time a model needs to be trained or tested. Instead, NumPy arrays can be saved as 'npy' files to Google Drive and later loaded back into Python. It only takes 14 minutes to load 1500 npy files with pixel dimensions 960 × 640 into Python.

TensorFlow offers a simpler and more efficient data format, the TFRecord, that can be easily incorporated into code for object detection [46]. TFRecords combine a bounding box label CSV file with its corresponding images to form a single record. Since all the data is stored in one place, the record takes less time to load into Python than individual images [35]. TFRecords for the full image set used a total of 0.4 GB of space while the NumPy arrays take over 4 GB. Therefore, the code in this study mainly used the three TFRecords corresponding to the training, validation, and test sets. Then a smaller set of NumPy arrays were used in custom code to visualise and evaluate the model performance on the validation and test sets. After labelling and storing the images in the correct formats, they were used to train a CNN.

Results and discussion

Met data analysis

There were two to seven coupons in each of the Met's images. The majority of the images contain six coupons, two of each metal type. In total, there are 12,964 coupons in the Met's images. Therefore, drawing and exporting the bounding box labels was a highly time-intensive process. In R, the box labels were compiled to find the percentage of the coupons of each metal type and corrosion level, as shown in Fig. 2. Silver and copper were most likely rated as permanent, while lead was more likely to be temporary. As noted in the literature [22], silver was rarely classified as unsuitable.

The Met images were randomly split into training, validation, and test sets with 70%, 20%, and 10% of the images respectively (1540, 448 and 220 images). Each of these image sets have approximately the same percentage of coupons at the different corrosion levels as in Fig. 2. This ensures that each class will be represented proportionally during the training, validation, and testing of the model.

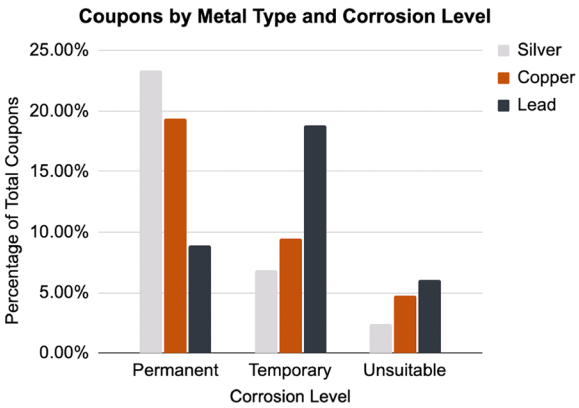


Fig. 2 Percentage of the total coupons falling in each of the metal and corrosion classes from the Met's dataset of 2208 images with 12,964 coupons

Training CNNs

This section describes the process of training, validating, and selecting a machine learning model with in-depth consideration of the model performance based on common metrics.

Since image processing requires a lot of memory and computing power, CNNs are trained with batches of images. For example, in one training step, a model can be fed a batch of four training images. The CNN weights are updated with each step. For a training set with 1540 images, it takes 385 steps to complete all the batches (which is called an epoch). Ideally a network will be trained for many epochs so each training image is seen multiple times.

The CNN will output three prediction components that can be compared to the ground truth bounding box labels. It outputs a predicted bounding box, a corresponding class, and a score. The score is the probability that the box contains an object with the predicted class, which can be considered the confidence the model has in the prediction.

Model metrics

Several metrics were used to evaluate the performance of the CNN. First, the intersection over union (IoU) represents the level of overlap between two boxes on a scale of 0 to 1, as shown in Fig. 3 [47]. An IoU of 1 would mean that the CNN located the object of interest perfectly. To evaluate the model, an IoU threshold, such as 0.5, can be set so that only predictions with IoUs greater than the threshold would be accepted.

IoU is used to classify a prediction as either a true positive (TP), false positive (FP), or false negative (FN). In Fig. 3, consider the object with a ground truth class of

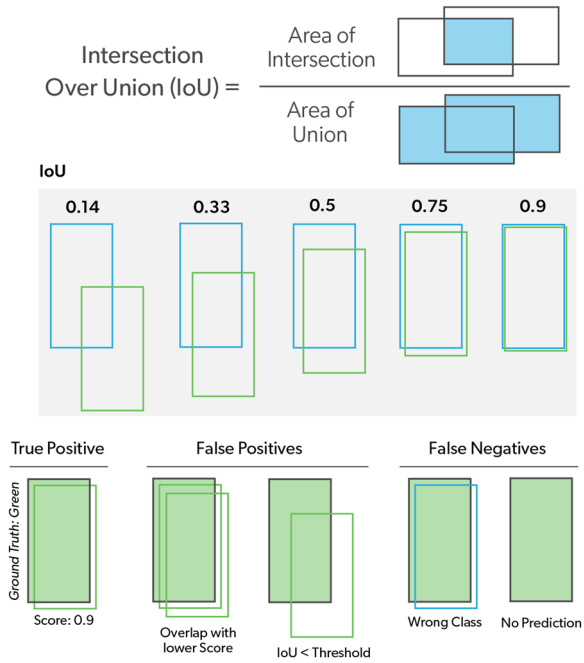


Fig. 3 The formula for Intersection over Union and some examples of True Positives, False Positives, and False Negatives

green. TP, FP, and FN are assigned for all the ground truth objects and predictions with class green. True positives are good predictions; they pass the IoU threshold, have the same class as the ground truth, and have the highest scores. False positives are predictions that are below the IoU threshold, including boxes with a poor fit or boxes that don't identify an object at all [47]. FPs also include predictions that overlap with previous boxes; they satisfy the IoU threshold, and have the same class as the ground truth, but don't have the highest scores. A false negative is any ground truth box that does not have any predictions of the correct class [47].

The total counts of TPs, FPs, and FNs are used to calculate the precision and recall for a particular class. Precision equals the number of true positives divided by the sum of true and false positives. Precision quantifies whether the model produces relevant boxes, as lots of overlapping or irrelevant boxes will create more FPs. Recall equals the total true positives divided by the sum of true positives and false negatives. Recall measures whether the model found all objects of the desired class. The average precision (AP) for a class equals the area underneath the precision-recall curve [48]. Mean average precision (mAP) is the mean of the average precisions for all of the classes. AP and mAP range from 0 to 1, or 100%, and values close to 1 theoretically indicate that a model performs well [47]. AP and mAP will be discussed further in the "Investigating differences in MAP" section.

Selecting a model

When choosing a CNN model, there is a trade-off between speed and precision. The first model tested in this study was the SSD ResNet50 V1 FPN 640x640 (RetinaNet50) [49]. RetinaNet50 is one of the faster models in the model zoo, and it had a decent mAP of 34.3% on the COCO dataset [40]. While this model is quick to train, there are more precise models. Of the models for smaller size images, CenterNet HourGlass 104 512×512 [50] had the highest mAP of 41.9% on the COCO dataset [40]. However, in practice it was extremely slow to train. As shown in Table 3, CenterNet completed less than half the epochs that RetinaNet50 did within roughly the same period of time. A compromise is the model EfficientDet D1 640×640 [44], which is faster than CenterNet and more precise than RetinaNet. EfficientDet D1 has the second-highest mAP of 38.4% for small image models [40]. After these initial trials, training these models on fewer epochs, the model EfficientDet D1 640x640 (EffDet) was selected. Generally, CNN models can learn more useful patterns from images if they complete more epochs during training. Therefore, after an initial test with EffDet version 1 (V1) that trained for 40 epochs, EffDet version 2 (V2) was trained for 100 epochs. Further research could explore more models, but the rest of this paper will focus on these two versions of the EffDet model.

Customising model configs

TensorFlow models are based on a model configuration (config) with many parameters that should be updated to fit the data. For example, some object detection models predict bounding boxes based on pre-defined anchors which suggest width-height ratios for the boxes [35]. The default anchor sizes are 0.5, 1.0, and 2.0 in the EffDet config, but these may not be suitable for different kinds of objects [51]. The width-height ratios of the coupon bounding boxes were calculated from the label data in R. For both versions of EffDet, the anchor sizes were updated to match the size of the different coupons present in the dataset: 0.3 for the long vertical coupons, 0.7 for shorter vertical coupons, and 3.0 for long horizontal coupons.

Another way to customise the config is to update the non-max suppression hyperparameters. Non-max suppression is a method of post-processing to reduce the number of outputted bounding boxes [35]. Two important parameters to adjust are the maximum number of detections per class and maximum total number of detections. For both EffDet models, the models will output up to 9 detections per class, equal to the maximum number of coupons of a particular class times the number of anchors (3 × 3). Then the maximum total detections is set to 81, the number of classes times the maximum

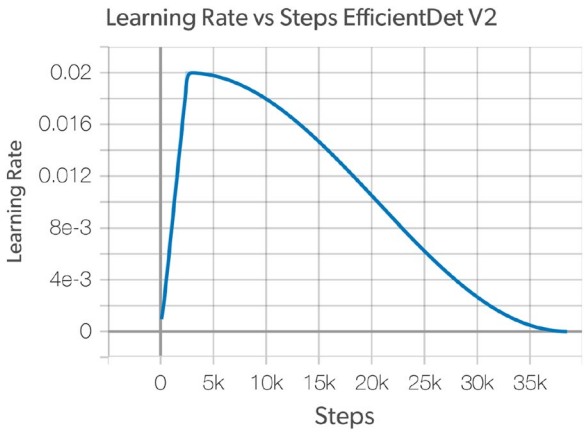


Fig. 4 Cosine decay learning rate in EffDet model version 2

number of detections per class (9 × 9) [51]. Versions 1 and 2 of EffDet start to differ with the non-max suppression score and IoU thresholds shown in Table 4. Version 1 has a score threshold that is approximately zero, so coupons with extremely low scores will still be outputted. Version 2 has a higher score threshold, so fewer bounding boxes will be outputted. The IoU threshold sets the level of IoU where a bounding box will be considered as an overlapping box [51]. The IoU threshold will be discussed further in the "Investigating Differences in mAP" section.

For smaller datasets, data augmentation can improve the accuracy of CNNs [37]. TensorFlow configs enable many different kinds of augmentation, from random crops to brightness adjustments [51]. Since the Met images were usually arranged in the same position, the CNNs in this paper used image flips to randomise the position of the coupons. Version 1 used only random horizontal flips while version 2 used both horizontal and vertical flips.

In order for the model to effectively learn image features, the parameters in the optimiser need to be updated. Optimisers such as stochastic gradient descent (SGD) are algorithms used to update the weights within a neural network. The optimiser's learning rate controls how quickly the model adjusts the weights to improve the performance. If the learning rate is too small, the model will only make small adjustments and take too long to train. Large learning rates make faster improvements in model performance, but if the rate is too large, the model may not settle on suitable weights. One way to address this is to decrease or 'decay' the learning rate over time to reach the optimal weights [35].

The EffDet config has a default of SGD with a momentum of 0.9 and cosine decay learning rate [44]. Fig. 4 shows how the learning rate in EffDet was linearly increased in a 'warm-up' period before decreasing according to cosine decay. As shown in Table 4, for both versions of EffDet, the learning rate started at approximately 0.00015 and then took 2500 warm-up steps to reach the specified learning rate bases. Based on the initial performance of the models, the config was adjusted so that version 1 and 2 have slightly different learning rate bases, shown in Table 4. The complete configurations of the EffDet models are provided in a GitHub repository [42].

Training and validation performance

The performance of the model was measured with a loss function. The optimiser SGD updates the model's weights in the direction that will reduce the loss [35]. While training the CNN, the loss on the training set was calculated at each step. The type of loss calculated was specified in the model configs. The loss should converge to a lower value, as shown in Fig. 5. In Tensorflow, the loss can also be calculated on the validation set by saving a periodic checkpoint, which is a file with the weight values from a certain stage of the model. Checkpoints were saved every two epochs (770 steps), which is why the validation line is smoother than the training line in Fig. 5. The aim was to see a decrease in both the training and validation loss [43]. Links to interactive loss plots from TensorFlow's visualisation tool, TensorBoard, are available in GitHub [42].

Despite the increased training time and parameter adjustments, the mean average precision values for version 2 of EffDet were lower than version 1, as shown in Table 5. $\mathbf{mAP}_{0.5}$ and $\mathbf{mAP}_{0.75}$ were the mean average precisions with IoU thresholds of 0.5 and 0.75. **mAP**

Fig. 5 Total loss from training and validation (eval) of EffDet version 2. The validation loss was calculated every 770 steps which equals two epochs

was the average of the mAPs at the IoU thresholds from 0.5 increasing to 0.95 by 0.05 [52]. Table 6 shows the average precision for each class. Again, version 2 has a lower average precision across all classes. Although mAP is an important metric for model performance, it is also abstract. The next section will investigate whether the lower mAP values for version 2 would really indicate that it is a poorer model than version 1.

Investigating differences in mAP

This section investigates the differences in mAP between the two EffDet models and suggests methods of improving mAP.

The mAP values in Table 5 were calculated with the COCO API metrics package [52] while the values in Table 6 were calculated with another mAP package available on GitHub [53]. The latter package also creates plots with counts of the predicted true and false positives. As shown in Fig. 6, version 1 created approximately 4000 bounding boxes for each of the classes. This is roughly equivalent to the maximum number of detections per class times the number of images in the validation set ($448 \times 9 = 4032$). By contrast, version 2 outputs fewer false positives, which is likely due to the increased score threshold in the non-max suppression step. In total, version 1 had 2,608 true positives while version 2 outputted 2571. Both were close to the total of 2621 ground truth coupons in the images. However, the difference of 37 true positives did not seem congruent with the large difference in mAP values at IoU 0.5 as shown in Tables 5 and 6.

When the IoU threshold was increased to 0.75, the number of true positives for version 1 dropped to only 1864 true positives. Meanwhile, version 2 outputted a more reasonable 2398 true positives. However, as shown in Table 5, at a 0.75 IoU threshold, the mAP value of version 1 was still greater than version 2. Since precision equals the number of true positives divided by the sum of true and false positives, why would version 2 have a lower mAP when it produced more true positives and fewer false positives than version 1?

To illustrate this paradox, consider the Met's image in Fig. 7 which only has two lead coupons, both rated temporary. For this image, version 1 outputted 72 predictions, including some Pb-T boxes with scores up to 0.88. However, the model also predicted false positives for every other corrosion and metal type with scores nearing zero. Since there are no true positives for these 8 classes, the average precision was only calculated for Pb-T [53]. Thus, 87% of version 1's predictions were

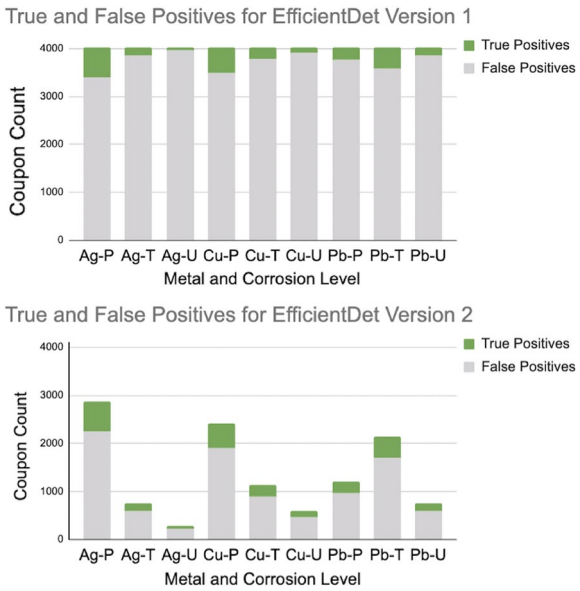


Fig. 6 Counts of true and false positives predicted by both EffDet models on the validation set at an IoU threshold of 0.5

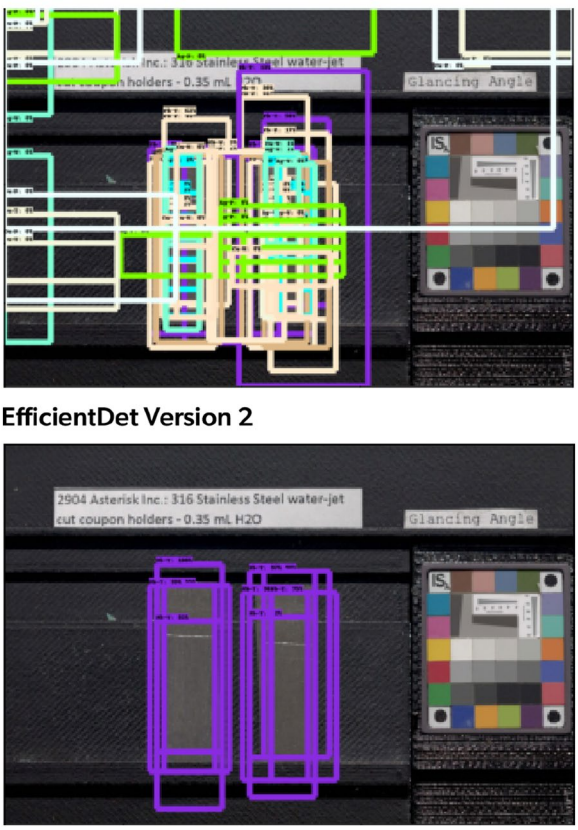


Fig. 7 Examples of an image with only two lead coupons (Pb-T) with all of the predicted bounding boxes from version 1 and 2 of EffDet. Met sample number 2904

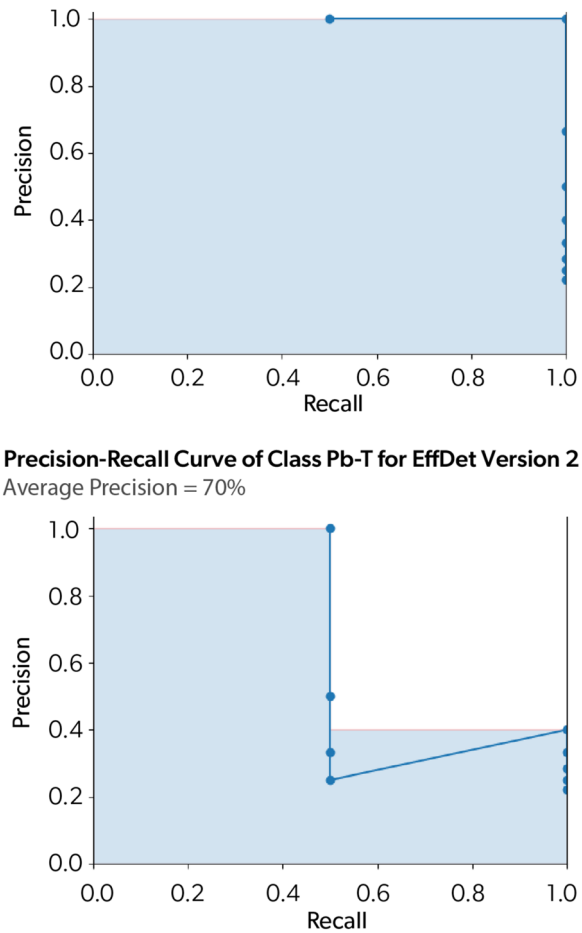


Fig. 8 Precision-Recall curves of class Pb-T for EffDet versions 1 and 2 from the predictions on the image shown in Fig. 7

Table 1 An example of the information given in the Met's image spreadsheet from sample number 1550

Metal	Results	Comments
Ag	P	Orange speckle on inward top
Cu	U	Reddening; hazing; black spots
Pb	T	Darkening; light violet film

irrelevant but not accounted for in the average precisions of this image.

Version 2 only outputted 9 bounding boxes, all of which were classified as Pb-T with IoU's greater than 0.5 for the ground truth coupons. This was likely due to both the increase in non-max suppression score threshold and the longer training time. However, the ranking of the scores of these 9 boxes caused the precision-recall curve to drop, as shown in Fig. 8.

The points of the AP curve were based on the rolling calculation of precision and recall. These calculations started with the highest scoring predicted bounding box, and then one box was added at a time in decreasing score order. The final precision and recall point was calculated with all of the outputted bounding boxes. This made the score-based order of the outputted boxes critical to the average precision. Since version 1's top two scoring predictions were true positives for the left and right coupons, its average precision equals 1. On the other hand, the first few highest scoring boxes from version 2 are for the coupon on the left. In this case, one true positive is followed by multiple false positives since they overlap with the first prediction. Then the next box was a true positive for the coupon on the right, so the precision rose again. Therefore, the mAP for version 2 will improve if there are fewer overlapping boxes.

One way to improve mAP for version 2 could be to lower the IoU threshold for the non-max suppression step [51]. The IoU threshold was used to suppress overlapping boxes. Since the coupons were tightly placed side-by-side in the images, the IoU threshold was raised to 0.7 in EffDet version 2. This means that only boxes with an IoU of at least 0.7 with a true positive box would be considered an overlap. If this threshold was decreased to 0.5 or lower, version 2 should theoretically output fewer overlapping coupons [51]. This should be tested in further research.

For the application to Oddy testing, the priority was to output high-quality predictions. Therefore, mAP overpenalises the overlapping boxes in version 2 while underpenalising the irrelevant predictions in version 1. As shown in the following section, it is possible to demonstrate that version 2 is the preferable model for detecting corrosion in Oddy tests by visualising the model outputs.

Visualising model outputs

One of the best ways to visualise the effectiveness of a trained model is to display the bounding box outputs on the validation and test set images. Usually, visualisation code outputs a certain number of boxes based on their scores [54]. As shown in the 'Before' picture of Fig. 9, drawing the top 10 highest scoring boxes can result in overlaps and missed coupons. Therefore, custom code was created to output only one prediction per ground truth coupon.

First, for each ground truth box, its IoU was calculated with each of the predicted bounding boxes. If a predicted box had an IoU above a certain threshold (the values 0.6 and 0.7 were tested), then it was included as a possible match for the ground truth coupon. After

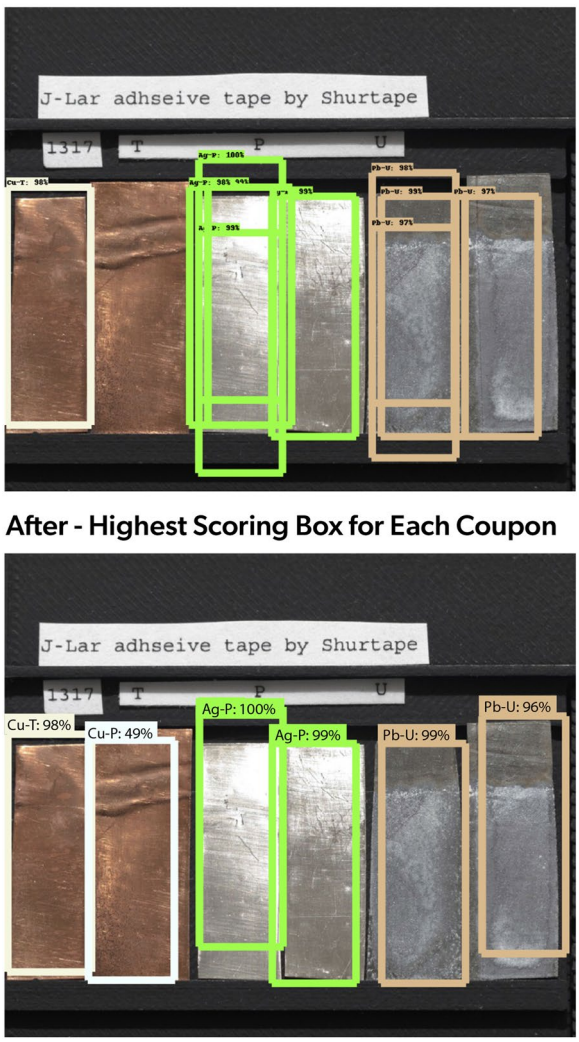


Fig. 9 Examples of boxes with standard outputted predictions and the outputs after custom code performed on Met images

compiling the IoU matches, the predicted box with the highest score was chosen as the model's best output for that ground truth box. As shown in Fig. 9, this method can effectively output clearer predictions. It can also help identify errors, such as the copper temporary coupon being misclassified as permanent.

These refined outputs can be transformed to calculate the percentage of coupons that were correctly identified in each of the validation images. As shown in Table 7, the EffDet models overall performed well with the majority of images having 100% correctly predicted coupons. However, version 2 of EffDet outperformed version 1 in multiple ways.

With an IoU threshold of 0.6, version 2 had 43 images with some level of incorrect predictions compared to version 1 with 58 images (9.5% vs. 13% of the validation set respectively). Version 1 also had two images where none of the predicted bounding boxes met the IoU thresholds for plotting. When the IoU threshold was raised from 0.6 to 0.7, version 1 performed markedly worse, with 108 of the validation images missing one or more coupon predictions. Version 2 was much more robust to the increase in IoU. This indicates that overall the predicted boxes from version 2 had a greater overlap with the ground truth boxes than in version 1. Despite having a lower mAP, version 2 of EffDet created more correct predictions than version 1 for the validation set. Therefore, only EffDet version 2 was used to analyse the performance on the test set and examine the model errors.

Performance on the test set

The mAP for the test set was 0.426 on average across IoU thresholds, 0.716 for a threshold of 0.5, and 0.487 for a threshold of 0.75. Table 8 shows the average precisions by class. There were higher APs on the test set compared to the validation set for all classes apart from Pb-T and Pb-U.

Table 9 shows that all images in the test set had at least 50% of the coupons correctly classified by EffDet version 2. Again, there was an increase in errors when the IoU threshold is raised from 0.6 to 0.7. The following section investigates the errors found in the predictions for both the validation and test sets at an IoU threshold of 0.6.

Error analysis

There were 43 images from the validation set and 19 images from the test set that had errors. Bounding boxes were drawn on these images using the method from the "Visualising model outputs" section at an IoU threshold of 0.6, and they were saved for further inspection. There were roughly the same number of glancing angle and side angle photos in the error images (33 and 29 respectively). This indicates that overall the model learned features well from images under both of the Met's lighting conditions. However, there were examples of errors that may have been caused by issues with lighting or image focus. One image in the validation set had the permanent copper coupons misclassified as unsuitable, but the image looks particularly dark. There was also an image where one Cu-U coupon was misclassified as Cu-P, but the image seems inordinately blurry. In order to make a more universal system for Oddy testing, future CNNs should be trained and tested with images with greater variation of lighting and levels of blurriness.

There are a few images in the Met's dataset that have a lead coupon lying horizontally across the top of some

other coupons in the image. There were two images in the test set and one image in the validation set where this coupon was not suitably detected by the CNN. Although data augmentation was used to make horizontal and vertical flips, most coupons are shown in parallel. Coupons with large overlaps and varying directions are a special case to addressed in further research. Additional data augmentation such as random rotations and brightness adjustments could improve model performance as well.

A contingency table of the predicted and ground truth coupon corrosion levels in the test set is shown in Fig. 10. Again, the majority of the coupons were correctly predicted. Most of the errors were contained within the larger boxes on the diagonal, indicating that the vast majority of the predicted metal types were correct. The model performed extremely well on silver coupons, but more errors were seen in copper and lead coupons. There are only a few errors where a coupon was predicted to be the wrong metal type. In one test set example, the copper coupon had thick white and blue corrosion that obscured most of the coupon's orange colour. In another, the unsuitable lead coupon had orange corrosion that the CNN misclassified as copper. Overall, the small amount of error in the coupon predictions indicates that the CNN was able to successfully learn features in the Oddy test data.

From the validation and test sets, 606 of the 668 images had correct predictions for all coupons. However, it is important to understand how coupon-level errors affected the overall classification of a test material so that the CNN can enable preventive conservation decisions. Fig. 11 shows contingency tables for the validation and test sets. There were 425 and 212 images with correct overall classifications for the validation and test sets respectively (95 and 96% of the total). Of the 43 validation images with errors in the coupon predictions, only 23 of those images had errors that affected the overall classification of the Oddy test. For the 19 test set images with coupon-level errors, only 8 of those had errors that affected the overall classification.

There are two types of errors in these classifications. The first, marked in amber in Fig. 11, is less harmful in the context of Oddy testing. These errors occurred when the predicted result was worse than the ground truth. There, a test material that could be acceptible for the display or storage of collections may not be classified as passing. The more concerning error shown in red occurs when the predicted value is more optimistic than the ground truth. In this case, a material that should be temporary or unsuitable could pass the Oddy test and potentially damage a museum object. The percentage of red errors is 2.22% and 2.27% for the validation and test sets respectively. Thus, approximately 2 out of every

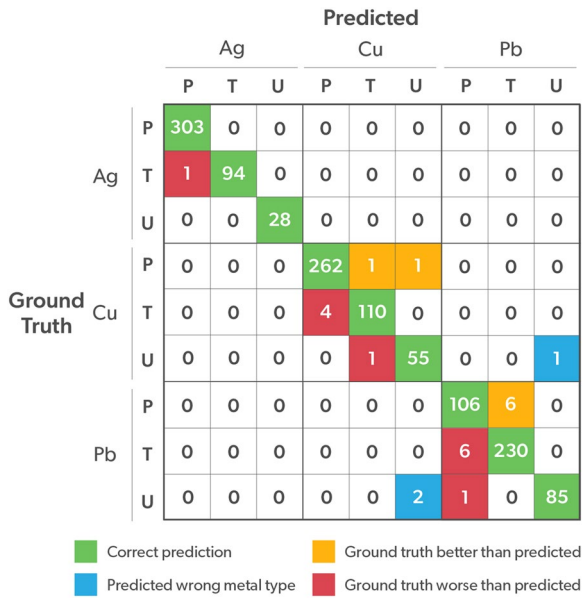


Fig. 10 Contingency table for the model's classifications for the coupons in the test set images at an IoU threshold of 0.6

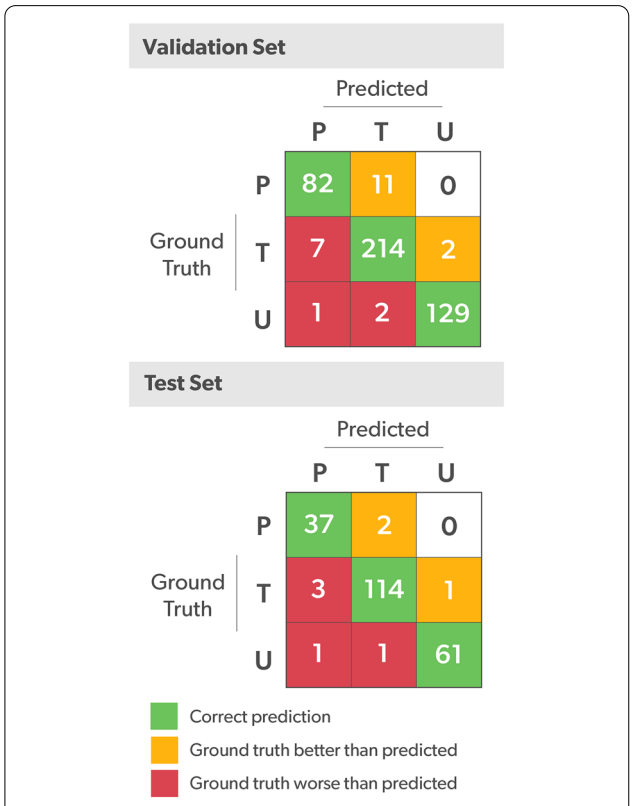


Fig. 11 Contingency table for the model's overall classifications of the test materials as permanent, temporary, or unsuitable. Calculated at an IoU threshold of 0.6

100 Oddy tests encountered by the CNN will result in a potentially damaging prediction. Future research should consider setting an appropriate error threshold as a goal for CNN performance, potentially using an assessment of the error level of human classifications as a guide.

Due to the subjective nature of the Oddy test, the original bias from the test evaluators at the Met is transferred from the data into the training of the CNN. While the CNN then may be similarly biased, since it has learned from thousands of images and multiple evaluators, it can reduce the individual bias in the classifications of corrosion. The biases in the data should be investigated and mitigated in further research, potentially by using images from more institutions. Nevertheless, the CNN's strong performance on the Met's data indicates that a CNN could improve the objectivity of the Oddy test, given images of a sufficient image quality and consistency.

Results from AIC wiki data

To obtain a separate dataset of images taken with other imaging protocols, experts from heritage institutions around the UK rated the coupons from the AIC wiki images. The experts were given the images and the overall rating of suitability from the wiki. Experts were not prompted to use any particular classification guidelines, so they rated each coupon based on their own experience and knowledge. Some experts rated the coupons in all 170 images while others completed a smaller batch of around 30 images. There were between 3 and 5 expert responses per image, as shown in Table 10.

To analyse this data, the expert ratings were compiled from their separate CSV files into one spreadsheet. In R, each coupon was assigned the mode of the coupon ratings from the experts. There were 21 coupons with ties and 18 inconclusive results. In the inconclusive results, many coupons had been rated as all three permanent, temporary, and unsuitable options by different experts. For ties, the images, comments from experts, and overall ratings provided in the wiki were re-examined to try to select a rating. In total, there were only 108 images with reasonable levels of agreement between the expert coupon ratings and the overall image rating from the wiki. Of those, there are only 18 images where the experts unanimously agreed on the coupon ratings.

These varying results align with the feedback from the experts that it was difficult to rate the images. The experts did not have access to the physical coupons or comments about the coupon conditions, so they were completely reliant on the images. Many instances of questionable image quality made the wiki image coupons challenging to rate. For example, as noted by some experts, silver is hard to photograph because it is particularly reflective. In many images from NYUL, the silver is so overexposed or reflective that it almost appears white. Experts tended to rate these coupons as permanent, which generally aligned with the overall image rating. However, this amount of reflection could obscure the information needed to identify the corrosion level with a CNN.

The CNN should only be used on tests that follow the Oddy test protocol. Particularly, coupons should not be in contact with the test material. There is one image from CMA where the adhesive test material was stuck on top of the coupons. Similarly, the coupons in the Autry image showed two different levels of corrosion because one half of each of the coupons were in contact with the test material. There was also clear tape over the coupons which could obscure corrosion information. While contact coupons are not good candidates for the CNN, an interesting case is coupons laid on top of test materials for the purpose of documentation. The CMA provided 48 images with fabric test materials underneath the coupons. These would be useful to improve the CNN's ability to detect coupons on different coloured backgrounds.

Given the variations in image quality and disagreement in ratings, the data from the AIC wiki is not yet sufficient to train an object detection CNN. However, this data is a good opportunity to test the limits of the Met data CNN.

Testing Met CNN on AIC data

Bounding boxes were drawn on the 18 wiki images with unanimous expert ratings. In total, there were 71 ground truth coupons. These were converted to npy files and inputted into the CNN to get bounding box predictions. The CNN performed very poorly out of the context of the Met's data. After applying the visualisation code, only 8 images had at least one suitable bounding box prediction, even with a reduced IoU threshold of 0.4. Only two Cu-U

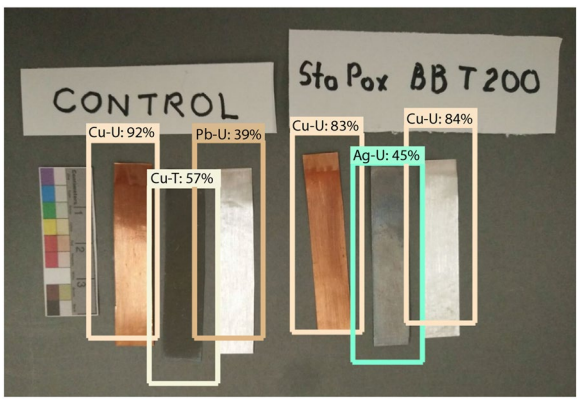


Fig. 12 Bounding boxes with predicted classes and scores from the Met CNN on an image from Heritage Conservation Centre Singapore. All of the coupons should be rated permanent

Table 2 Examples of the format needed for object detection image labels

Width	Height	Class	Xmin	Ymin	Xmax	Ymax
960	640	Cu-U	17	242	129	553
960	640	Ag-P	246	264	351	555
960	640	Pb-T	458	243	570	556

Each row corresponds to a different bounding box. The columns 'width' and 'height' are the image dimensions. These examples correspond to the Met sample 1550 with GA lighting

Table 3 The training times, epochs, and batch sizes for the models tested from TensorFlow's model zoo [40]

Model	Time	Epoch	Batch
RetinaNet50 [49]	1h 40m	50	10
CenterNet HourGlass104 512x512 [50]	1h 43m	20	5
EfficientDet D1 640x640 - V1 [44]	2h 30m	40	4
EfficientDet D1 640x640 - V2 [44]	6hrs 9m	100	4

Table 5 The mean average precision for the models on the validation set

Model	mAP	$mAP_{0.5}$	$mAP_{0.75}$
EffDet - V1	0.554	0.925	0.567
EffDet - V2	0.422	0.692	0.500

These calculations were from the COCO API on GitHub [52]. The subscript numbers represent IoU thresholds, and **mAP** was averaged across IoU thresholds from 0.5 increasing to 0.95 by 0.05

coupons and one Pb-U coupon from the HOD images were correctly classified.

The Met coupons are relatively large and take up a solid proportion of the image. The NYUL coupons, however, are very small, so the CNN was unable to produce suitable predictions for them. Future CNNs should be trained on images with coupons over a range of size scales. Similarly, as demonstrated in Fig. 12, the bounding boxes did not fit well to the coupons for HCC images. These coupons had different width-height ratios than the Met coupons, so adding more anchor sizes to the CNN config could help. The CNN also classified the coupons as the wrong corrosion levels and metal types.

Having experts rate unfamiliar images from other institutions was a highly flawed method. However, the CNN needs more image variety to detect and assess coupons from a wide variety of Oddy test and image capture procedures. Thus, future research should collect images directly from more institutions. Gathering data from multiple sources will lead to a more robust and generalised model for a universal system.

Table 4 The parameters that differ between the model configs for version 1 and 2 of the EffDet models

Parameter	Version 1	Version 2
Score threshold	9.99999993922529e-09	0.2
IoU threshold	0.5	0.7
Initial learning rate	0.00015384616	0.00015384616
Learning rate base	0.012307692	0.02
Total steps	15400	38500

Conclusion

While alternatives to Oddy testing are proposed often, it is still widely used and trusted globally by stewards of cultural heritage. This paper explored the use of an object detection CNN to reduce the subjectivity in determining the level of corrosion in Oddy tests. The CNN utilises the information learned from thousands of images to create predictions that may be less biased than a human examiner. The CNN correctly detected 98% of coupons in a dataset of images captured at the Met. Factoring in the coupon-level errors, 96% of the test set images had the correct overall image classifications of permanent, temporary, or unsuitable. Therefore, this research has demonstrated that object detection can be effective for classifying corrosion levels with relatively few errors.

Further research could more methodically test the CNN options by training different models for the same

Table 6 The average precisions (AP) by class for EffDet V1 and V2 at an IoU threshold of 0.5 on the validation set

AP	EffDet V1 (%)	EffDet V2 (%)
Ag-P	96.39	72.94
Ag-T	94.18	75.31
Ag-U	95.41	76.30
Cu-P	96.30	60.37
Cu-T	87.20	58.96
Cu-U	89.72	72.65
Pb-P	86.96	58.03
Pb-T	93.69	72.60
Pb-U	94.36	76.97
mAP	92.69	69.35

The mAP values differ slightly from Table 5 because they were calculated using another mAP package [53]

Table 7 The count of validation set images with a certain percentage of correctly predicted coupons after the custom code reduced the number of outputted boxes. ED stands for EffDet

	IoU > 0.6		IoU > 0.7	
Correct (%)	**ED V1**	**ED V2**	**ED V1**	**ED V2**
0	3	2	5	2
17	2	1	1	1
20	3	-	-	-
25	-	-	3	-
33	-	2	2	2
50	10	6	12	6
57	-	-	1	-
67	15	16	34	16
71	-	-	-	1
75	4	1	3	1
83	20	14	112	22
86	1	1	-	-
100	390	405	275	397

Table 8 Average precision by class and metal type on the test set with IoU > 0.5 for EffDet V2

	Ag (%)	**Cu (%)**	**Pb (%)**
P	74.30	61.08	63.00
T	84.35	61.94	71.48
U	85.01	75.79	68.94

Table 9 Performance of EffDet V2 on the 220 images in the test set

Correct (%)	**IoU > 0.6**	**IoU > 0.7**
50	1	2
67	7	6
83	9	16
86	2	2
100	201	194

number of epochs and adjusting their configuration (config) parameters. Using the Python notebooks developed for this research, more models from the TensorFlow model zoo can be easily tested. For an EfficientDet model, the non-max suppression IoU threshold should be lowered. If this does not improve the mAP and produce fewer overlapping bounding boxes, then further configuration adjustments should be considered. Ideally future model predictions will also have higher IoU values with the ground truth boxes. EffDet V2 was evaluated at a threshold of 0.6, but IoU's greater than 0.7 would produce boxes with a tighter fit around the coupons. This could potentially be achieved by adding more anchor sizes. To choose a final model, the mAP values and the prediction errors should be compared.

Table 10 The number of images used from each source in the AIC wiki

Source	**Total Images**	**Agreed**	**Unanimous**	**Experts**
Autry	1	0	0	5
CMA	65	45	1	3-5
HCC	44	27	5	3-4
HOD	26	11	2	3-4
NYUL	34	25	10	4
Total	170	108	18	-

'Agreed' is the image count where expert coupon ratings were overall in agreement. 'Unanimous' is the count of images where the experts unanimously agreed on coupon ratings

Given these promising results, the long-term aim of this research is to create a tool so that the Oddy test CNN is publicly accessible and universally applicable. However, it was not effective to have experts rate images from other institutions. Instead, researchers should reach out to preventive conservators and heritage scientists to create a database of labeled image data. The images gathered should ideally represent a range of lighting scenarios, blurriness, coupon sizes, and background colours. Even a hundred images from a few institutions could build a helpful dataset to train a model. For example, the next CNN could be trained on images from institutions that also use the Met's protocol. Researchers should also establish an error threshold for the type of CNN error that is potentially damaging to museum objects with the participating institutions.

If institutions can not provide coupon-level data, researchers could consider training a classification CNN. The classification CNN would give an overall rating of permanent, temporary, or unsuitable for an image. These CNNs would also require less researcher time because the images do not need to be labelled with bounding boxes. However, the coupon-specific information from an object detection CNN could help testers learn more from the model's output.

In the long-term, this CNN model can be the basis of a publicly accessible web tool. The tool should aim to reduce the time needed to evaluate an Oddy test, help train people who are learning how to run Oddy tests, and improve the overall objectivity. Certainly more research is necessary to create a final product that can be incorporated into museum workflows around the world. Nonetheless, this project created over 12,000 bounding box

labels and many Python notebooks that are ready to be reused in future research. Although experts should make the final important conservation decisions, the CNN can summarise the corrosion information from thousands of Oddy test images to aid the work of experts. Thus, this research shows that the state-of-the-art method of object detection CNNs is a promising solution to reduce the subjectivity in Oddy tests.

Abbreviations

CNN: Convolutional Neural Network; AIC: American Institute for Conservation; Met: The Metropolitan Museum of Art; Ag: Silver; Cu: Copper; Pb: Lead; GC-MS: Gas chromatography-mass spectrometry; PVC-U: Unplasticized polyvinyl chloride; GA: Glancing angle; Si: Side angle; P: Permanent; T: Temporary; U: Unsuitable; API: Application programming interface; IoU: Intersection over union; config: configuration; SGD: stochastic gradient descent; EffDet: EfficientDet D1 640x640.

Acknowledgements

Thank you to the scientists who rated the images from the Oddy Test AIC wiki data including Fabiana Portoni, Tatiana Marasco, Keeley Wilson, Constantina Vlachou, David Thickett, and Paul Lankester.

Author contributions

EL completed this work as a part of their master's dissertation and JGB supervised. EL developed the method, labelled images for the CNN, trained the machine learning model, and analysed the results. AB and EB provided data from the Metropolitan Museum of Art, and AB rated the ratings of individual coupons in a spreadsheet. DT organised the rating of the AIC wiki images. JGB, AB, EB, and DT provided feedback and guidance on the research. All authors read and approved the final manuscript.

Funding

Not applicable,

Availability of data and materials

The Oddy Test image dataset analysed during this study is available in the AIC Wiki repository [33]. The images from the Metropolitan Museum of Art analysed during the current study are available from the authors on reasonable request. The code and datasets generated and analysed during the current study are available in the following GitHub repository [42], https://github.com/emilyrlong/oddy-test/.

Declarations

Ethics approval and consent to participate

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Institute for Sustainable Heritage, University College London, London, UK. [2]Victoria and Albert Museum, London, UK. [3]Metropolitan Museum of Art, New York, NY, USA. [4]English Heritage, London, UK.

Received: 21 June 2022 Accepted: 17 August 2022
Published online: 27 September 2022

References

1. Oddy WA. An unsuspected danger in display. Mus J. 1973;73:27–8.
2. Stephens CH, Buscarino I, Breitung E. Updating the oddy test: comparison with volatiles identified using chromatographic techniques. Stud Conserv. 2018;63(S1):425–7.
3. Korenberg C, Keable M, Phippard J, Doyle A. Refinements introduced in the Oddy test methodology. Stud Conserv. 2018;63(1):2–12.
4. Bamberger JA, Howe EG, Wheeler G. A variant oddy test procedure for evaluating materials used in storage and display cases. Stud Conserv. 1999;44(2):86–90.
5. Robinet L, Thickett D. A new methodology for accelerated corrosion testing. Stud Conserv. 2004;48(4):263–8.
6. Torok E, Wickens JDJ. Reevaluating the Oddy Test: An Examination of the Diversity in Protocols Used for Material Testing in the United States. In: Conservation and Exhibition Planning: Material Testing for Design, Display, and Packing. Washington, DC: Smithsonian American Art Museum & National Portrait Gallery; 2015. p. 33. https://fdocuments.net/document/conservation-and-exhibition-planning-material-testing-for-4-though-we-often.html. Accessed 17 May 2022.
7. Heine H, Jeberien A. Oddy test reloaded: standardized test equipment and evaluation methods for accelerated corrosion testing. Stud Conserv. 2018;63(S1):362–5.
8. Shen J, Shen Y, Xu F, Zhou X, Wu L. Evaluating the suitability of museum storage or display materials for the conservation of metal objects: a study on the conformance between the deposited metal film method and the Oddy test. Environ Sci Pollut Res Int. 2018;25(35):35109–29.
9. Green LR, Thickett D. Testing materials for use in the storage and display of antiquities-a revised methodology. Stud Conserv. 1995;40(3):145–52.
10. American Institute for Conservation. Oddy Test Protocols; 2020. https://www.conservation-wiki.com/wiki/Oddy_Test_Protocols. Accessed 22 Aug 2021.
11. ISO. ISO 23404:2020 Information and documentation—Papers and boards used for conservation—Measurement of impact of volatiles on cellulose in paper. ISO/TC 46/SC 10; 2020. https://www.iso.org/standard/75439.html. Accessed 6 Jun 2022.
12. Zhang J, Thickett D, Green L. Two tests for the detection of volatile organic acids and formaldehyde. J Am Inst Conserv. 1994;33(1):47–53.
13. Newman R, Derrick M, Byrne E, Tan M, Chiantore O, Poli T, et al. Strange Events Inside Display Cases at the Museum of fine arts, Boston, and lessons to be learned from them—part 1. In: Conservation and exhibition planning: material testing for design, display, and packing. Washington: Smithsonian American Art Museum & National Portrait Gallery; 2015. p. 11. https://aiccm.org.au/wp-content/uploads/2020/01/materialtesting conference-2015-abstractbooklet2FEA0168725A.pdf
14. Hatchfield P, Goppion S, Chiantore O, Poli T, Riedo C, Suslick K, et al. Strange Events Inside Display Cases at the Museum of fine arts, Boston, and lessons to be learned from them—part 2. Beyond the oddy test - the way forward. In: Conservation and exhibition planning: material testing for design, display, and packing. Washington: Smithsonian American Art Museum & National Portrait Gallery; 2015. p. 12–13. https://aiccm.org.au/wp-content/uploads/2020/01/materialtestingconference-2015-abstractbooklet2FEA0168725A.pdf
15. Alvarez-Martin A, George J, Kaplan E, Osmond L, Bright L, Newsome GA, Kaczkowski R, Vanmeert F, Kavich G, Heald S. Identifying VOCs in exhibition cases and efflorescence on museum objects exhibited at Smithsonian's National Museum of the American Indian-New York. Herit Sci. 2020;8(1):1–13.
16. van Iperen J, van Keulen H, Keune K, Abdulah K, van Langh R. Crystalline deposits in new display cases at the rijksmuseum: characterisation and origin. Stud Conserv. 2021;66(5):253–71. https://doi.org/10.1080/00393630.2020.1811475.
17. Samide MJ, Smith GD. Assessing the suitability of unplasticized Poly(Vinyl Chloride) for museum showcase construction. J Am Inst Conserv. 2020;61:1–13.
18. Samide MJ, Liggett MC, Mill J, Smith GD. Relating volatiles analysis by GC-MS to Oddy test performance for determining the suitability of museum construction materials. Herit Sci. 2018;6(1):1–10.
19. Thickett D. Frontiers of preventive conservation. Stud Conserv. 2018;63(S1):262–7.
20. Stephens CH, Breitung EM. Impact of volatile organic compounds (VOCs) from acrylic double-sided pressure-sensitive adhesives (PSAs) on metals found in cultural heritage environments. Polym Degrad Stab. 2021. https://doi.org/10.1016/j.polymdegradstab.2021.109738.
21. Thickett D. Oxygen depletion testing of metals. Heritage. 2021;09(4):2377–89. https://doi.org/10.3390/heritage4030134.

22. Buscarino IC, Stephens CH, Breitung EM. Oddy test protocol at the Metropolitan Museum of Art (The Met); 2021. https://www.conservation-wiki.com/w/images/9/94/20190226_OT_1_MMA_Oddy_Protocol.pdf. Accessed 10 Aug 2021.
23. Samide A, Stoean R, Stoean C, Tutunaru B, Grecu R, Cioateră N. Investigation of polymer coatings formed by polyvinyl alcohol and silver nanoparticles on copper surface in acid medium by means of deep convolutional neural networks. Coatings (Basel). 2019;9(2):105.
24. Samide A, Stoean C, Stoean R. Surface study of inhibitor films formed by polyvinyl alcohol and silver nanoparticles on stainless steel in hydrochloric acid solution using convolutional neural networks. Appl Surf Sci. 2019;475:1–5.
25. Du J, Yan L, Wang H, Huang Q. Research on grounding grid corrosion classification method based on convolutional neural network. MATEC Web Conf. 2018;160:01008.
26. Cha Y, Choi W, Suh G, Mahmoudkhani S, Büyüköztürk O. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. Comput Aided Civ Infrastruct Eng. 2018;33(9):731–47.
27. Katsamenis I, Protopapadakis E, Doulamis A, Doulamis N, Voulodimos A. Pixel-level Corrosion Detection on Metal Constructions by Fusion of Deep Learning Semantic and Contour Segmentation. In: International Symposium on Visual Computing 2020: Advances in Visual Computing; 2020. p. 160–169.
28. Yao Y, Yang Y, Wang Y, Zhao X. Artificial intelligence-based hull structural plate corrosion damage detection and recognition using convolutional neural network. Appl Ocean Res. 2019;90: 101823.
29. Bastian BT, NJ, Ranjith SK, Jiji CV. Visual inspection and characterization of external corrosion in pipelines using deep neural network. NDT & E International: Independent Nondestructive Testing and Evaluation. 2019;https://doi.org/10.1016/j.ndteint.2019.102134.
30. Wang N, Zhao X, Zhao P, Zhang Y, Zou Z, Ou J. Automatic damage detection of historic masonry buildings based on mobile deep learning. Autom Constr. 2019;103:53–66.
31. Zou Z, Zhao X, Zhao P, Qi F, Wang N. CNN-based statistics and location estimation of missing components in routine inspection of historic buildings. J Cult Herit. 2019;38:221–30.
32. Lamas A, Tabik S, Cruz P, Montes R, Martínez-Sevilla Á, Cruz T, et al. MonuMAI: dataset, deep learning pipeline and citizen science based app for monumental heritage taxonomy and classification. Neurocomputing. 2021;420:266–80.
33. American Institute for Conservation. Oddy Test Results: Combined Results; 2020. https://www.conservation-wiki.com/wiki/Combined_Materials_Testing_Results. Accessed 20 Aug 2021.
34. Hodgkins R, Centeno S, Bamberger J, Tsukada M, Schrott A. Silver nanofilm sensors for assessing daguerreotype housing materials in an oddy test setup. e-Preserv Sci. 2013;01(10):71–6.
35. Zafar I, Tzanidou G, Burton R, Patel N, Araujo L. Hands-on convolutional neural networks with tensorflow. 1st ed. Birmingham: Packt; 2018.
36. Skalski P. Make Sense; 2019. https://www.makesense.ai. Accessed 7 Aug 2022.
37. Shorten C, Khoshgoftaar TM. A survey on image data augmentation for deep learning. J Big Data. 2019;6(1):1–48.
38. Google. Welcome to colaboratory; 2021. https://colab.research.google.com/notebooks/intro.ipynb. Accessed 16 Aug 2021.
39. Huang J, Rathod V, Sun C, Zhu M, Korattikara A, Fathi A, et al. TensorFlow object detection API; 2020. https://github.com/tensorflow/models/tree/master/research/object_detection. Accessed 12 Aug 2021.
40. TensorFlow. TensorFlow 2 detection model Zoo; 2020. https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md. Accessed 19 Aug 2021.
41. COCO. COCO 2017 Object detection task; 2017. https://cocodataset.org/#detection-2017. Accessed 12 Aug 2021.
42. Long, ER. oddy-test: Convolutional neural network to detect corrosion in oddy tests; 2022. https://github.com/emilyrlong/oddy-test/. Accessed 6 Feb 2022.
43. Abu-Mostafa YS. Lecture 13: validation. California Institute of Technology; 2012. http://work.caltech.edu/slides/slides13.pdf. Accessed 10 Apr 2021.
44. Tan M, Pang R, Le QV. EfficientDet: Scalable and Efficient Object Detection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020. p. 10778–10787. https://doi.org/10.1109/CVPR42600.2020.01079.
45. TensorFlow. Introduction to tensors; 2022. https://www.tensorflow.org/guide/tensor. Accessed 27 Aug 2021.
46. Oliveira D. Creating TFRecords. Keras. 2021; https://keras.io/examples/keras_recipes/creating_tfrecords/. Accessed 20 Aug 2021
47. Tan RJ. Breaking down mean average precision (mAP); 2019. https://towardsdatascience.com/breaking-down-mean-average-precision-map-ae462f623a52. Accessed 27 Aug 2021
48. Padilla R, Passos WL, Dias TLB, Netto SL, da Silva EAB. A comparative analysis of object detection metrics with a companion open-source toolkit. Electronics. 2021. https://doi.org/10.3390/electronics10030279.
49. Lin T-Y, Goyal P, Girshick R, He K, Dollar P. Focal loss for dense object detection. IEEE Trans Pattern Anal Mach Intell. 2020;42(2):318–27.
50. Duan K, Bai S, Xie L, Qi H, Huang Q, Tian Q. CenterNet: Keypoint Triplets for Object Detection. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV); 2019. p. 6568–6577.
51. Morgunov A. TensorFlow Object Detection API: Best practices to training, evaluation & deployment. Neptune; 2021. https://neptune.ai/blog/tensorflow-object-detection-api. Accessed 21 Aug 2021.
52. cocodataset. COCO API. GitHub; 2020. https://github.com/cocodataset/cocoapi. Accessed 19 Aug 2021.
53. Cartucho J, Ventura R, Veloso M. Robust object recognition through symbiotic deep learning in mobile robots. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); 2018. p. 2336–2341. https://github.com/Cartucho/mAP. Accessed 22 Aug 2021.
54. TensorFlow. Object detection; 2022. https://www.tensorflow.org/hub/tutorials/object_detection. Accessed 20 Aug 2021.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com