



Achieving remanufacturing inspection using deep learning

Chigozie Nwankpa, et al. *[full author details at the end of the article]*

Received: 20 September 2019 / Accepted: 24 September 2020 / Published online: 15 October 2020
© The Author(s) 2020

Abstract

Deep learning has emerged as a state-of-the-art learning technique across a wide range of applications, including image recognition, object detection and localisation, natural language processing, prediction and forecasting systems. With significant applicability, deep learning could be used in new and broader areas of applications, including remanufacturing. Remanufacturing is a process of taking used products through disassembly, inspection, cleaning, reconditioning, reassembly and testing to ascertain that their condition meets new products conditions with warranty. This process is complex and requires a good understanding of the respective stages for proper analysis. Inspection is a critical process in remanufacturing, which guarantees the quality of the remanufactured products. It is currently an expensive manual operation in the remanufacturing process that depends on operator expertise, in most cases. This research investigates the application of deep learning algorithms to inspection in remanufacturing, towards automating the inspection process. This paper presents a novel vision-based inspection system based on deep convolution neural network (DCNN) for eight types of defects, namely pitting, rust, cracks and other combination faults. The materials used for this feasibility study were 100 cm × 150 cm mild steel plate material, purchased locally, and captured using a USB webcam of 0.3 megapixels. The performance of this preliminary study indicates that the DCNN can classify with up to 100% accuracy on validation data and above 96% accuracy on a live video feed, by using 80% of the sample dataset for training and the remaining 20% for testing. Therefore, in the remanufacturing parts inspection, the DCNN approach has high potential as a method that could surpass the current technologies used in the design of inspection systems. This research is the first to apply deep learning techniques in remanufacturing inspection. The proposed method offers the potential to eliminate expert judgement in inspection, save cost, increase throughput and improve precision. This preliminary study demonstrates that deep learning techniques have the potential to revolutionise inspection in remanufacturing. This research offers valuable insight into these opportunities, serving as a starting point for future applications of deep learning algorithms to remanufacturing.

Keywords Inspection in remanufacturing · Deep learning for inspection · Automated inspection deep learning in remanufacturing

Introduction

Remanufacturing is a crucial strategy for achieving environmental conscious manufacturing and product recovery [1, 22], which brings about sustainable development through product reuse and associated energy and material savings [30, 31]. It is an industrial process by which used products and their sub-units known as cores are brought to an 'as good as new' functional condition with matching or better warranty [32, 51]. According to the US Institute for Trade and Commerce, major products currently being remanufactured include automobile parts, heavy-duty equipment, industrial machines, aerospace equipment, electrical and electronic products, photocopiers, tyres, IT and medical equipment [63].

As an industrial process, remanufacturing is made up of the following three independent operations [43]:

- Disassembly
- Processing which includes processes such as cleaning, inspection, testing and other operations that takes the used products to an 'as new' condition.
- Reassembly.

Remanufacturing is ultimately a product recovery technique, and it differs from other end-of-life approaches as it returns products that are as good as new ones; while creating new business opportunities, jobs, and increasing product affordability [18, 59]. Moreover, remanufactured products typically save up to 90% of materials compared to newly manufactured ones and consume just about one-sixth of energy compared to new products [48, 59].

In the developed world, remanufacturing helps businesses to keep up with regulations such as End-of-life vehicle (ELV) Directive and the Waste Electrical and Electrical Equipment (WEEE) Directive [14, 66] that are aimed at controlling environmental pollution and wasteful consumption habits. Under the ELV legislation, the original equipment manufacturer (OEM) becomes responsible for the cost incurred by the end-of-life (EOL) treatment of cars. This allows the last owner to take the vehicle to dispose of it at an authorised EOL treatment facility at no expense. The legislation further requires OEMs to make their products more amenable to dismantling, increase reuse and minimise the use of hazardous materials. Similarly, the WEEE directive requires users to discard their WEEE at no cost, at a central collection centre provided either by the OEM or in conjunction with the local council. Again, the OEM has a price to bear in this arrangement. However, by implementing remanufacturing, manufacturers can potentially make profits while keeping with the requirements of these regulations [9, 14].

Although the value of remanufacturing is unknown, a recent report has put the actual cost of the automotive remanufacturing industry alone to worth \$138billion by 2020 [42]. Other vast remanufacturing industry sectors, including the aerospace and electronics industries, add to the immense potentials in remanufacturing while recognising that it still has great potential to grow further. However, much of the anticipated growth is hindered by critical issues, including the very complicating nature of the remanufacturing process. Crucial matters outlined in the literature include [23, 48]:

- Difficulties in achieving the adequate supply of used products or cores as well as new parts to serve as a replacement.
- Challenges in developing efficient and effective remanufacturing process tools and techniques to carry out efficient disassembly, inspection and cleaning and without difficulty.

- Mistrust and poor customer perception of remanufactured products, usually because of lack of knowledge and awareness of the capabilities of remanufacturing.

This study focuses on the aforementioned second class of issue and specifically, inspection tools and techniques. It examines eight common types of faults that are found in the automotive remanufacturing industry, which includes combinations of crack, rust, and pitting surface faults. We trained a deep neural network using the obtained fault datasets, which are made up of 28,800 image samples. The sample split of training and validation was performed, where the training sample was used to train the network and evaluated based on the validation dataset and live video feed.

The remaining parts of this paper are structured in sections, where section 2 presents an overview of inspection in remanufacturing. Section 3 presents the general deep learning-based systems and highlights the architectures of deep neural networks; most importantly, the convolutional neural networks used in the design. Section 4 outlines the design approach for the deep learning vision inspection system including the data collection, pre-processing, choice of model, training and evaluation of the model. Section 5 details the implementation of the developed model alongside the setup. Section 6 outlines the performance evaluation results with section 7, presenting the conclusion and future works.

Inspection in remanufacturing

In remanufacturing, inspection is a necessary process used to guarantee quality and reliability [35]. Consequently, inspection cells are often integrated into a functional remanufacturing workshop floor [1] to facilitate core classification and to determine the extent of deterioration and quality [13]. Inspection reduces the risk of economic loss associated with carrying out early stages of the remanufacturing process on cores that are to be discarded eventually for not being of acceptable quality. Effective inspection techniques have been found to increase the remanufacturer's profit, satisfy customers [21] and mitigate risks associated with uncertainty in core supply and quality [56].

A minimum of three primary stages of inspection is usually carried out in a typical remanufacturing process: pre-disassembly (core) inspection, post disassembly inspection of components and final inspection of the whole system following reassembly. The main objective of core inspection is to determine the economic and feasibility of remanufacturing a core as well as information required for planning the remanufacturing process, including inventory planning [33, 60]. For instance, this may be aimed at determining the core's design model, quality or identification of some indicators that point to its remanufacturability [70]. Post disassembly inspection, on the other hand, is conducted on components or sub-assemblies after disassembly to determine whether they could be reused or not. This inspection also includes other forms of examination performed during the remanufacturing process alongside after reassembly process of the whole product before the final testing before putting the matching warranty. The stages of inspection in remanufacturing are shown in Fig. 1, with the focus areas specified by the grey clouds.

Efficient information management is vital to remanufacturing inspection; scholars [17, 39] demonstrate the value of information management in inspection and in addressing the uncertainty in remanufacturing. They show that information relevant to core inspection may be managed by radio frequency identity (RFID) tagging and by keeping an accurate record of other information gathered manually about cores.

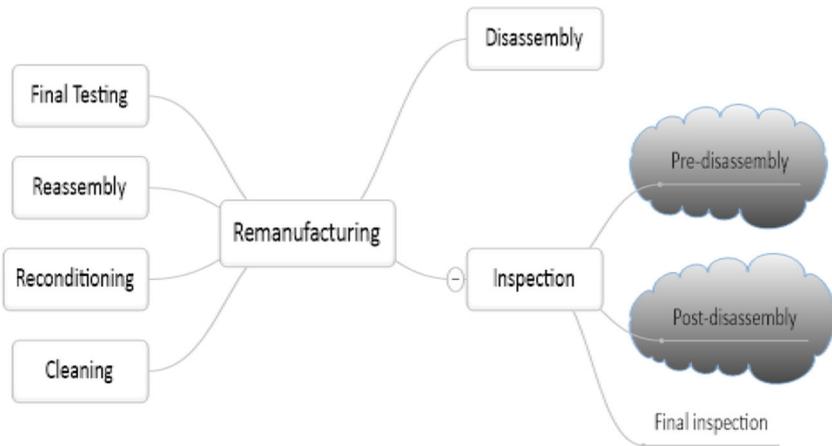


Fig. 1 Remanufacturing process block diagram showing the focus inspection stages

The use of T method-3 pattern recognition technique to identify features capable of enhancing the pre-processing inspection of the automotive crankshaft has been demonstrated [67] as one of the first inspection approaches. Similarly, Zhang et al. [69] proposed a metal magnetic memory (MMM) inspection technique for predicting the residual life of structural cores. The authors demonstrate the potential of MMM in detecting macro and micro-crack as well as in predicting the residual useful life of a core, which is necessary to ensure that reused cores do not have inherent faults that would cause premature product failure. However, this method detects only crack defects, which makes it limited to achieve automated inspection alone without incorporating other techniques to identify different types of faults.

A recent inspection approach using FUZZY Technique for Order Preference by Similarity to Ideal Situation (FUZZY TOPSIS) has been proposed by [46]. The TOPSIS approach is amongst the useful techniques for ranking and selecting several possible alternatives using Euclidean distance measurement. Its principle is based on the assumption that the chosen option will have the shortest distance from the ideal positive solution and the longest distance from the perfect negative solution. The inspection model uses part dimensions such as length, surface roughness, height, groove thickness, diameter and other parameters to optimise the selection of used engine piston for remanufacturing. However, this approach is complicated, as most of the returned product information is not available by default, which makes it challenging to adopt in the remanufacturing industry where there are various products of different dimensions.

The requirements of existing inspection technologies potentially complicate the inspection process; making a new and more effective solution expedient. Hence, the development of new systems and algorithms that could similarly view the world, as humans perceive it, becomes a research approach for improving inspection in remanufacturing. These fields often referred to as computer vision and machine learning has been applied in numerous areas with state-of-the-art results in image analysis, classification, recognition, video analysis, natural language processing and even recommender systems and many others.

Gibbons et al., (2018) recently attempted the use of image data for modelling remanufacturing inspection, to achieve automated visual inspection for remanufacturing. They proposed the use of the Gaussian mixture model to train a machine learning-based inspection model for visual corrosion inspection of high-value components [16]. Though this method was successful in categorising corroded and non-corroded areas of tested engines, it was only able to detect

corrosion faults, which makes it unsuitable for holistic inspection in remanufacturing; thereby calling for more robust approaches that could detect multiple defect types at the same time.

Perhaps, recent advances in computer vision research have opened new frontiers of the capabilities of emerging technologies with deep learning, making the headlines in almost all fields of applications. The use of deep learning algorithms has opened new areas of research, focussing on the implementation of deep learning to a different problem. Among the applications is the use of deep learning for remanufacturing inspection, which we explore in the study. It proposes a vision-based inspection system, which can contribute to a real-time, non-contact, reliable, faster, cost-effective remanufacturing inspection, thereby automating the inspection process. Conversely, the deep learning-based inspection system can be applied at the pre-disassembly inspection stage, to sort Eol products into remanufacturable and scraps, as well as to the post-disassembly stage where Eol products are inspected during any of the stages of the remanufacturing process which include cleaning, reconditioning or even final testing stage.

However, this research focuses mainly on the post-disassembly inspection stage, where most often, disassembled products are merged during the cleaning process, thereby prompting the need for automated inspection and sorting process. The developed deep learning-based inspection system becomes a handy technology to automatically check the Eol product conditions and classifies them into similar products.

Deep learning based systems

The deep learning-based inspection systems are computational models that use multi-layered neural networks, stacked together with additional layers to extract features used to describe patterns in data. These stacked neural networks are characterised by their makeup as architectures of deep learning found in different deep learning research.

There are many architectures of deep learning which include the recurrent neural networks [12], the deep belief networks [49], the deep autoencoders [65] and the convolutional neural networks [19]. These architectures are developed and tested against challenging problems using the standard datasets of ImageNet, MNIST, CIFAR10 and COCO datasets [34, 36, 45, 64], to obtain the automatic deep learning-based features used to ascertain the performance of the architectures including the convolutional neural networks.

These deep learning architectures have unique properties, which are highly related to the type of data they work on. The architectures include the deep belief networks, which are used to describe statistical models and techniques, represent sets of variables and their conditional dependencies [26, 49]. Also, the deep autoencoder architectures are mostly used to copy inputs to outputs [61, 65]. In contrast, the recurrent neural networks are useful for modelling sequential data [12, 40], as well as the CNN for modelling recognition and classification problems because they perform well especially in processing grid of values [19, 52]. This property of the convolutional architectures makes them suitable for analysing image data from the computer vision inspection problems.

Convolutional neural networks

The convolutional neural networks (CNNs) is one of the vital architectures used in deep learning research. This architecture is inspired by the natural mechanism of visual perception of living creatures [29] and has been successfully applied in image recognition, object

detection and localisation [24, 40, 62]. The CNN is a type of feed-forward neural network, that consist of three main layers, which include the convolutional layers, pooling layers, and fully connected layers. They are arranged by multiple alternating convolutional and pooling layers, followed by the fully connected layers. The CNN architectures use the convolution and pooling operations to extract low to higher levels of features in the input data. The CNN has been used mostly for image-based analysis [54, 57] although the CNN has found other application areas which include biological, health, face and finger recognition systems [15, 44, 55]. These layers are explained as follows.

The CNN comprises of the input layer, which accepts the input to the network. It takes the presented images and performs some pre-processing of the input, especially resizing the input images, before passing them to the subsequent layers for feature extraction.

The convolutional layers are used to obtain feature maps of a given input image by applying some filters across a given local region. For a given network, with k number of filters, W^i as weights for i filters, and filter bias b^i , activation function σ , a small image patch x_s and input image size of $M \times N$, the convolution of filter i and input x_s is given by

$$f_{i s} = \sigma (W^i x_s + b^i) \quad (1)$$

By sliding the patch of size $a \times b$ across the image, and using a stride of size s , we obtain a convolution output given by [54]

$$k \times \left\lceil \frac{(M-a+1)}{s} \right\rceil \times \left\lceil \frac{(N-b+1)}{s} \right\rceil \quad (2)$$

Where $\lceil \cdot \rceil$ is a ceiling function.

Furthermore, the activation functions are generally applied to the output of the layer to convert the learned linear mappings to non-linear mappings for further propagation of the parameters of the networks. Examples of the common activation functions used in deep learning applications include the softmax, sigmoid, rectified linear units (ReLU), exponential linear units (ELU), and softplus [50]. The function tries to keep the CNN computation stable by ensuring that the learned parameter values do not overshoot towards infinity or get stuck around 0, thereby maintaining the specific desired range of values.

The pooling layers accepts the high-dimensional input images and down-samples their spatial dimensions. Pooling is applied conventionally after the convolution layers to reduce the learned features dimensionality and avoid overfitting of the network. The pooling operations can be either average pooling, sum pooling, max-pooling or global average pooling. However, most DCNN architectures use the max-pooling used in their design where the maximum value of each block is used as the corresponding pixel value on the output image. The max-pooling operation is given by [54].

$$\text{Maxpool} = \max(x_s) \quad (3)$$

Besides, max-pooling a $M \times N$ input image with a patch size of $a \times b$, we obtain an output, given by

$$\left\lceil \frac{(M-1)}{a} \right\rceil \times \left\lceil \frac{(N-1)}{b} \right\rceil \quad (4)$$

The fully connected layers make up the last layers of most CNN architectures, used to compute class scores using the softmax activation function [50]. It accepts the high-level filtered images

and transforms them into categorical labels. More recently, the convolutional layers have also been used in the last layers of the network architectures, to learn the feature maps in the data, end-to-end thereby removing the need to resize the input images as the network can learn from any size of image [47].

The first CNN used for recognition task was named LeNet5, developed and used for digit recognition [41] with just five layers. However, after the LeNet5 architecture, other architectures of CNN have evolved and increased in depth. The CNN breakthrough came with the development of the AlexNet architecture, which is composed of eight convolutional layers, with the pooling and other layers in the architectures [38]. Other CNN based architectures for object recognition include VggNet [58], SegNet [4], ResNet [25], and SeNet [27] model etc.

ResNet architecture

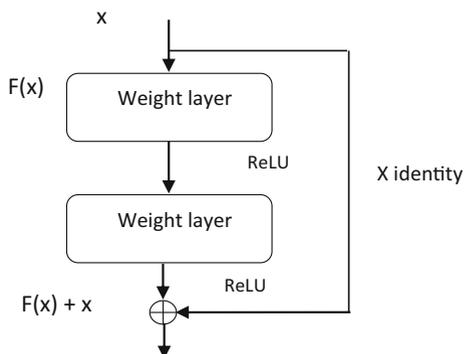
The ResNet architecture is a CNN architecture that uses identity shortcut connection to skip one or more layers during the training of the network. The ResNet allows stacked layers of the network to fit a residual mapping rather than fit the desired mapping directly [25]. The basic idea of the building block of the ResNet architecture is shown in Fig. 2.

The ResNet architecture emerged as the best, and first architecture for image recognition and image localisation applications in ILSVRC 2015 challenge and the architecture was very deep, surpassing a hundred layers deep in its design [25]. The ResNet architecture attained remarkable results of the top-5 error rate of 3.57%. The significant advantage provided by the identity shortcut connections is that they do not add any further parameters to the network as well as additional computational complexity. Besides, the deep residual networks proved that increased depth of the network provided higher accuracy.

The CNN architecture used for the development of this inspection system is the ResNet architecture. The choice of ResNet architecture is based on the need to achieve very high accuracy when we test the network. The ResNet has multiple architecture variants which include the ResNet18, ResNet34, ResNet50, ResNet101, and ResNet152 architectures [25]. These architectures have a different number of layers with the increasing number implying that the depth of the network is increasing, and the greater layers lead to improved recognition accuracy.

The first traditional vision-based inspection systems explored hand-crafted algorithms to extract the points of interest in data such as edges [8, 20] and corner detectors [2, 3], scale-invariant feature transform [11], speeded-up robust features [5], the histogram of oriented

Fig. 2 ResNet residual learning building block



gradients [10], and many other techniques. However, some other vital features considered when quantifying these component features of image data include shape, colour, and texture [6]. The deep learning approach of image classification is a unique contrast to the traditional image classification approach because it automatically learns features in data and quantifies the content of the data thereby saving us much time, needed to manually compute the features of the given image sample.

Model design approach

The deep learning-based vision inspection system for remanufacturing is an inspection technique that explores the use of neural networks to achieve inspection. The design and implementation of the deep learning-based vision inspection system are based on the real-time application of surface fault inspection. In this paper, we demonstrate the feasibility of classifying surface faults using the deep convolutional neural network approach. Also, the training of the designed network to achieve the DL vision-based inspection system was a crucial factor in the techniques used for data acquisition and processing.

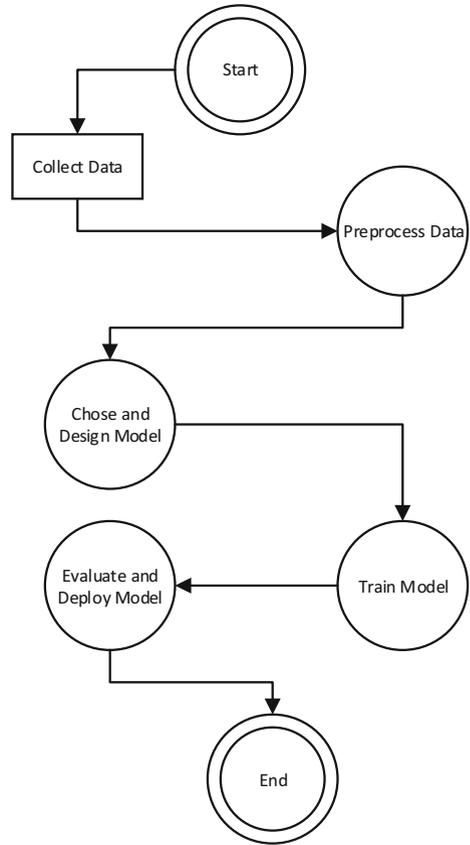
The deep learning-based inspection system is formulated as a supervised learning problem where the dataset of images of faulty components are captured, and their corresponding class labels are used to teach the deep learning classifier what each category looks like. If the classifier makes an incorrect prediction, we can apply other methods to correct the mistakes by telling it what the correct label is and this process continues until the specified stopping criterion is met. These criteria are set to monitor the performance of the network, which includes accuracy, the number of epochs that represents the total number of iterations of the input data and other inherent, considered factors. The process flow diagram for the model is shown in Fig. 3.

The model flow chart for the deep learning based classification problem can be categorised into key process component stages which include; data collection where the video of the defected parts are recorded. The pre-processing stage involves converting the recorded videos of the parts to images, useful for training and testing the model. The chose and design model stage involves selecting the appropriate computational model and modifying it to suit the proposed remanufacturing inspection application. Also, the training stage involves passing the training images of the parts to the model, with proper weights, biases and optimisation for effective feature extraction. The model evaluation stage entails passing the test set images to the model for classification. The performance of the designed system is obtained by recording the accuracy of the model's respective prediction on the presented image samples. The detailed procedure and explanations about the model development are outlined in the following sections.

Data collection

The data collection involved the use of a webcam USB camera of 0.3 megapixels, with lens F/2.0 and $f = 4$ mm, programmed to record three minutes of video data of the samples in a controlled lighting conditions. The video data was augmented to reflect some of the challenges of computer vision systems, which includes objects at different orientations, scale, differing lighting conditions.

Fig. 3 Model Development approach for the DL based vision inspection system



The videos consist of different fault conditions, with one good part without defect, and others comprising of various defects including pitting, rusting and crack defects. Also, the combination defects include rust and crack defects, pitting and crack defects, rust and pitting defects as well as rust, crack and pitting defects. The pitting defects are formed as a result of localised corrosion confined within a small area on the metal surface [53]. Besides, rusting is another surface defect that occurs when metals are exposed to air and moisture, thereby forming oxide of iron. Finally, the crack defect is another form of a severe flaw that commonly originate at the surface and grows with increasing stress [7]. These faults and their combinations form part of the majority of the typical surface faults that the visual inspection systems can detect, especially without very sophisticated camera devices. A total of seven defected parts and one good part making up the eight video samples recorded (Fig. 4).



Fig. 4 Images of some defected samples used in the model

Data preprocessing

This stage involves converting the videos into images, resizing and labelling each of the associated images. The categories require the seven identified types of defect that the algorithm will classify. The defect labels include; pitting surface faults (PF), crack surface faults (CF), rust surfaces faults (RF), pitting and crack surface faults (PnC), rust and crack surface faults (RnC), rust and pitting surface faults (RnP) and pitting, crack and rust surface faults (PnRnC) alongside sample non-defected surfaces (Nodef). Besides, the images were afterwards resized to suit the image size of the ResNet model.

The output of the pre-processing stage produced images, which consist of some 3600 images of each part and the eight classes produced the 28,800 images used to train and test the model. The data was split into 80% training, and 20% for testing and evaluate the deep learning-based vision inspection system. A total of 23,040 images were used to train the ResNet model, as well as 5760 images, for the model's performance evaluation.

Chose and design model

The stage involves choosing the right computational model that can perform the desired inspection task. There are several deep learning-based computational models, which have performed reasonably well on object recognition and detection problems. To select the appropriate model, a critical consideration was the accuracy of the model on recognition tasks and the ResNet architecture was chosen, based on the high accuracy of the model. ResNet was the first deep CNN architecture that surpassed human-level performance on image recognition problem [25]. The output of this model was modified to suit the number of classes in the data for training and inference.

Another design consideration is the inference speed, which is a concern, especially when there is a need to deploy these models on lower performance systems. After deployment, the model inference is performed on central processing units (CPU's), embedded systems and even mobile devices, thereby making the computational cost, a concern in the development of this system.

Train model

The training of the network is the most complicated stage of the deep learning-based vision inspection system. The training process involves multiple iterations over the training set to obtain the optimal parameters of the classifier. In the training stage, we are concerned about tuning the parameters that make our algorithm efficient. These parameters include setting the learning rate, number of epochs, decay rate, regularisation, dropout and many other parameters of the network. These parameters are referred to as hyperparameters, and they must be appropriately set for the adequate performance of the classifier.

In the training stage, the model obtains the appropriate features from the data by optimising the weights and biases, which is usually saved for future inference purposes. The training is performed using the algorithm of stochastic gradient descent optimiser where the algorithm tries to estimate the gradients using some single randomly picked group of training examples; thus, the parameters of the CNN are obtained. The learning process involves the forward pass and backpropagation of the gradients obtained from the connected model weights. A significant issue avoided during the gathering of these data is the class imbalance where there are

more examples of a type of fault, and this biases the algorithms during training, thereby causing overfitting. This problem was avoided by using the same number of training examples in all classes. However, researchers have outlined specific techniques to manage imbalanced dataset, including resampling the training set, k-fold cross-validation, quintuplet sampling, among others [28].

In our model, we have eight sample categories which imply that the output layer must match the total number of eight samples. The appropriate and suitable values of the hyperparameters of the network were determined through experience and experimentation. The hyperparameters set include learning rate, dropout, and mini-batch size, and drop factor. The hyperparameters of the systems were carefully chosen such that the model converges correctly during training as Krizhevsky, et al., (2012) highlighted that enormous batch sizes affects the quality of the final solution [37]. The total number of epochs, which is the number of times that the entire dataset samples are presented to the model, usually referred to as epoch, and other parameters were set. The optimised model parameters used to train the deep CNN model used in this research is shown in Table 1, highlighting the parameters used in the study.

Evaluate and deploy model

The evaluation stage involves presenting the separated test set images to the developed classifier and making predictions on the label of the image. The predicted models are compared against the ground-truth labels from our test set, with the ground-truth representing the actual image categories, and the number of correct predictions compiled. The performance of our network is quantified using standard performance metrics including the probability output and the confusion matrix to visualise the results.

The confusion matrix is a table used to describe the performance of a classification model on some specific test set for which their actual values are known. The matrix shows the comparison of the ground truth and the models' predictions as output.

Implementation of the system

The design of the DL vision inspection system is based on ResNet18 architecture. This design is a MATLAB based ResNet CNN architecture requires RGB input images of size [224, 224, 3] and consist of five each of sizes 64, 128, 256 and 512 filters, making up the 20

Table 1 shows the parameters of the designed model used in the research

Parameter Name	Parameter Values
Minibatch size	20
Learning rate	1e-3
Dropout	50%
Optimiser	Stochastic gradient descent with momentum
Loss function	Categorical cross-entropy
Number of epochs	4
Number of classes	8
Train/Test split ratio	7:3
Learning rate drop period	2
Learning rate drop factor	1e-4

convolutional layers found in the ResNet18 architecture, alongside one fully connected layer and a classification layer. The ResNet18 implementation has batch normalisation after each of the convolutional layers, alongside the ReLU activation function, which converts the linear weights to non-linear for further propagation. Also, the pooling layers specifically Max and Average pooling were used in the architecture with the Max pooling used at the first layer and Average pooling at the last convolutional layer, just before the fully connected layer. The standard mathematical addition function was also used to add the two branches of the architecture for each of the filter sizes used in the architecture. A softmax function was used to provide a probability output, for the classification layer, to give the results of the designed system [25].

Implementation setup

The block diagram of the DL based-vision inspection system is shown in Fig. 5, which highlights the three critical stages of the proposed method.

The input represents the set of videos and images captured from the connected cameras. The DCNN algorithm is the ResNet based CNN classifier, used to perform the inspection on the given objects. The output is a set of predictions made by the DCNN algorithm, which is the obtained deep learning vision-based inspection results. These output labels and the prediction probability scores are displayed as results of the developed system. The entirety of these blocks makes up the developed deep learning-based vision inspection system for remanufacturing application. Depending on the configurations, these outputs can be sorted into fault types, scraps, recondition and reuse, which are determined, depending on the nature of the Eol products, as well as the actuators used to sort the Eol products are they are conveyed.

The setup was achieved using a 2.7GHz, 8GB RAM computer with a GT740 graphics card and a web webcam USB high definition camera of 0.3 megapixels. The material 100 cm × 150 cm mild steel plate with different defects was recorded as videos at about 0.5 m distance from the camera. The camera was moved about to capture the data such that it reflects some of the critical issues facing computer vision systems which include scale, rotation invariance, poor lighting conditions, to mention a few. The movement of the camera reflects the augmentation needed for the designed system.

Performance evaluation and discussion

The ResNet architecture was chosen because of the accuracy of the architecture with Yosinki et al., outlining that if we have a small dataset, transfer learning is the best approach to best fine-tune the network for efficient learning [68]. To achieve this, the weights were frozen during the network training, achieved using transfer learning. The classification layers were also modified to suit the vision inspection dataset.

Furthermore, the training and evaluation of the designed system were performed in three stages, which include; training the model, evaluation using the validation data and using live

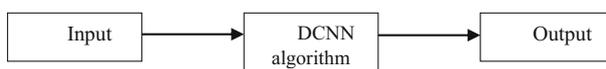


Fig. 5 Block diagram of the system implementation

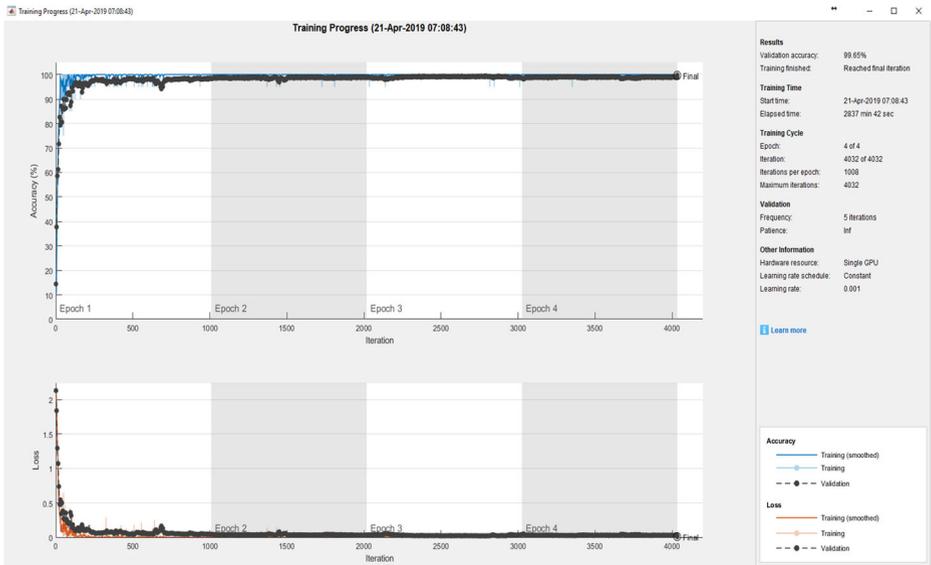


Fig. 6 Results of the training progress showing model prediction accuracy and loss

camera feed. Training and validation accuracies of 100% and 99.65% respectively were obtained on the designed system after training and testing of the model, with the results depicting almost an ideal model conditions. These ideal conditions are classification accuracy of 100% and loss of approximately zero. The Matlab based display of the training progress response of the design is shown in Fig. 6, showing the models accuracy and loss response.

The developed model confusion matrix is an 8×8 matrix, which represents the ground truth and the predicted output as the labels of the axes of the response. The validation data of 5760 images were used to test the system as shown in the confusion matrix results, which

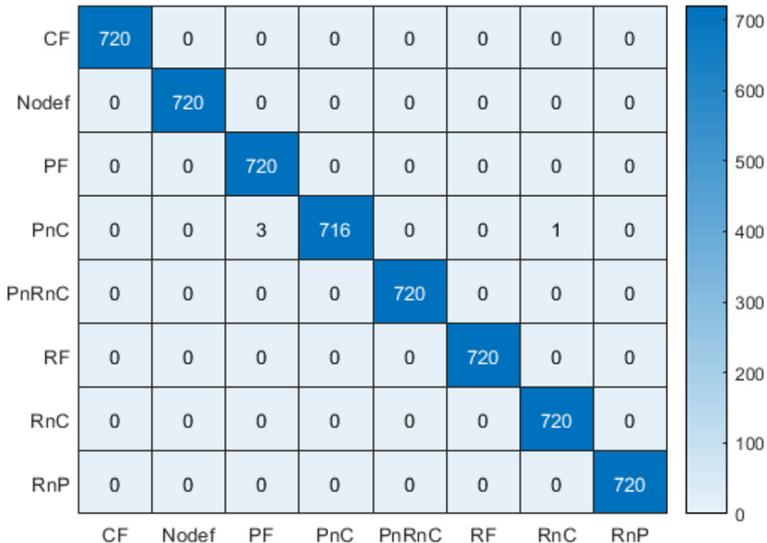


Fig. 7 Confusion matrix visualisation of the evaluation results of the DL based vision system

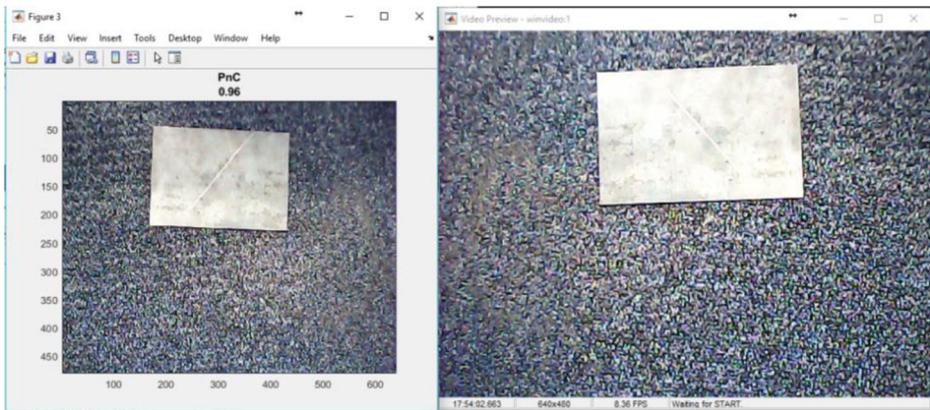


Fig. 8 Video Inspection results showing video preview and inspection results

outlined that three pitting and crack defects (PnC) were misclassified as pitting defects (PF) and the one PnC defect misclassified as rust and crack (RnC) defect. This result means that out of the 5760 predicts, 5756 were correctly predicted with four incorrect predictions. These results validate the high accuracy performance of the system, and it is shown in Fig. 7.

Conversely, the second stage of the testing involved the use of the video feed, which is the USB camera that captures the surfaces for inspection. The system processes the captured picture frames from the live camera video stream and performs the desired inspection, with the results displayed within a fraction of a second as shown in Fig. 8.

From the Fig. 8, the video preview of the inspection camera is shown on the right image with the model prediction of PnC (pitting and crack fault) with 96% accuracy on top-1 prediction, shown on the left-hand side of the figure.

Conclusion

In this paper, a DL vision-based inspection system is presented. The system adopted feature transfer from a pre-trained ResNet18 where we modified the network architecture to suit our dataset. This system takes one second to recognise and perform a vision-based inspection on a 2.7GHz CPU with NVIDIA GT 740 GPU. The results obtained from the MATLAB Deep Learning toolbox show that the proposed method performed exceptionally well at approximately 100% accuracy on the validation data shown in the confusion matrix. The high performance was sustained on testing with a live video feed from a USB camera.

By adopting this model, we can achieve automated inspection in remanufacturing especially to identify defected Eol product and sort them into predefined Eol options. It would enhance throughput, speed and reduce incidents of an industrial accident in remanufacturing, especially working with contaminated surfaces. This model aims to eliminate the sole dependence on operator judgment, which most remanufacturing companies depend on, to achieve precise inspection results.

The future work would be to implement the vision-based inspection with the most current state of the art deep learning architecture, to improve the accuracy of the results and compare the CNN architecture on industry-based datasets.

Acknowledgements The authors are grateful to Mr. Fuad Warsame, for assisting with the data capture process in the lab.

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/>.

References

1. Aksoy HK, Gupta SM (2005) Buffer allocation plan for a remanufacturing cell. *Comput Ind Eng* 48:657–677. <https://doi.org/10.1016/j.cie.2003.03.007>
2. Arrebola F, Camacho P, Bandera A, Sandoval F (1999) Corner detection and curve representation by circular histograms of contour chain code. *Electron Lett* 35:1065. <https://doi.org/10.1049/el:19990764>
3. Asada H, Brady M (1986) The curvature primal sketch. *IEEE Trans Pattern Anal Mach Intell PAMI-8*:2–14. <https://doi.org/10.1109/TPAMI.1986.4767747>
4. Badrinarayanan V, Kendall A, Cipolla R (2015) SegNet: a deep convolutional encoder-decoder architecture for image segmentation
5. Bay H, Ess A, Tuytelaars T, Van Gool L (2008) Speeded-up robust features (SURF). *Comput Vis Image Underst* 110:346–359. <https://doi.org/10.1016/j.cviu.2007.09.014>
6. Bharati MH, Liu JJ, MacGregor JF (2004) Image texture analysis: methods and comparisons. *Chemom Intell Lab Syst* 72:57–71. <https://doi.org/10.1016/J.CHEMOLAB.2004.02.005>
7. Broberg P (2013) Surface crack detection in welds using thermography. *NDT E Int* 57:69–73. <https://doi.org/10.1016/j.ndteint.2013.03.008>
8. Canny J (1986) A computational approach to edge detection. *IEEE Trans Pattern Anal Mach Intell PAMI-8*: 679–698. <https://doi.org/10.1109/TPAMI.1986.4767851>
9. Chen J, Hsu Y (2017) Competitive pricing and reusability choice for remanufacturable products. *Actual Probl Econ* 7:169–188
10. Dalal N, Triggs B Histograms of Oriented Gradients for Human Detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). IEEE, pp 886–893
11. Lowe DG (1999) Object recognition from local scale-invariant features. In: proceedings of the international conference on computer vision-volume 2 - volume 2. IEEE Computer Society, Washington DC, p 1150
12. Elman JL (1990) Finding structure in time. *Cogn Sci* 14:179–211. [https://doi.org/10.1016/0364-0213\(90\)90002-E](https://doi.org/10.1016/0364-0213(90)90002-E)
13. Errington M (2006) Remanufacturing inspection models
14. Fakhredin F, Bakker C, Geraedts J, Huisman J (2013) Five perspectives on Design for end of life: highlights of a literature review. In: *EcoDesign International Symposium*
15. Gao F, Yue Z, Wang J, Sun J, Yang E, Zhou H (2017) A novel active Semisupervised convolutional neural network algorithm for SAR image recognition. *Comput Intell Neurosci* 2017:1–8. <https://doi.org/10.1155/2017/3105053>
16. Gibbons T, Pierce G, Worden K, Antoniadou I (2018) A Gaussian mixture model for automated corrosion detection in remanufacturing. In: 16th international conference on manufacturing research ICMR. IOS press
17. Gildea LF, Adickes MD (1993) Front end shop floor control in a remanufacturing area. *SME*
18. Giutini R, Gaudette K (2003) Remanufacturing: the next great opportunity for boosting US productivity. *Bus Horiz* 46:41–48. [https://doi.org/10.1016/S0007-6813\(03\)00087-9](https://doi.org/10.1016/S0007-6813(03)00087-9)
19. Goodfellow I, Bengio Y, Courville A (2017) *Deep learning*. MIT Press, Cambridge
20. Griffith AK (1973) Edge detection in simple scenes using a priori information. *IEEE trans Comput C-22*: 371–381. doi: <https://doi.org/10.1109/T-C.1973.223724>
21. Gu Q, Gao T (2016) Management strategy of inspection error rate in reverse supply chain of remanufacturing. *Jisuanji Jicheng Zhizao Xitong/computer Integr Manuf Syst CIMS* 22:2469–2477. Doi: <https://doi.org/10.13196/j.cims.2016.10.022>
22. Gungor A, Gupta SM (1999) Issues in environmentally conscious manufacturing and product recovery: a survey. *Comput Ind Eng* 36:811–853. [https://doi.org/10.1016/S0360-8352\(99\)00167-9](https://doi.org/10.1016/S0360-8352(99)00167-9)

23. Hammond R, Amezcuita T, Bras B (1998) Issues in the automotive parts remanufacturing industry – a discussion of results from surveys performed among remanufacturers. *Int J Eng Des Autom – Spec Issue Environ Conscious Des Manuf* 4:27–46
24. He K, Zhang X, Ren S, Sun J (2015) Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification arXiv Prepr 150201852
25. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. *IEEE Conf Comput Vis Pattern Recognit* 2016:770–778. <https://doi.org/10.1109/CVPR.2016.90>
26. Hinton GE, Osindero S, Teh Y-W (2006) A fast learning algorithm for deep belief nets. *Neural Comput* 18: 1527–1554. <https://doi.org/10.1162/neco.2006.18.7.1527>
27. Hu J, Shen L, Albanie S, Sun G, Wu E (2017) Squeeze-and-excitation networks. *ArXiv Prepr* 170901507
28. Huang C, Loy CC, Tang X (2016) Learning deep representation for imbalanced classification. 2016 *IEEE Conf Comput Vis pattern Recognit*. doi: <https://doi.org/10.1109/CVPR.2016.580>
29. Hubel DH, Wiesel TN (1968) Receptive fields and functional architecture of monkey striate cortex. *J Physiol* 195:215–243
30. Ijomah WL, Childe S, McMahon C (2004) Remanufacturing: a key strategy for sustainable development. *3rd Int Conf des Manuf sustain Dev* 22:99–102. doi: <https://doi.org/10.1016/j.buildenv.2006.10.027>
31. Ijomah WL, Childe SJ, Hammond GP, McMahon CA (2005) A robust description and tool for remanufacturing: a resource and energy recovery strategy. In: 2005 4th international symposium on environmentally conscious design and inverse manufacturing. *IEEE*, pp 472–479
32. Ijomah WL, McMahon CA, Hammond GP, Newman ST (2007) Development of design for remanufacturing guidelines to support sustainable manufacturing. *Robot Comput Integr Manuf* 23:712–719. <https://doi.org/10.1016/j.rcim.2007.02.017>
33. Ilgin MA, Gupta SM (2010) Environmentally conscious manufacturing and product recovery (ECMPRO): a review of the state of the art. *J Environ Manag* 91:563–591. <https://doi.org/10.1016/j.jenvman.2009.09.037>
34. Imagenet (2018) Large scale visual recognition challenge (ILSVRC). In: *Standord Vis. Lab*
35. Kin STM, Ong SK, Nee AYC (2014) Remanufacturing process planning. *Procedia CIRP* 15:189–194. <https://doi.org/10.1016/J.PROCIR.2014.06.087>
36. Krizhevsky A (2009) Learning multiple layers of features from tiny images
37. Krizhevsky A (2014) One weird trick for parallelising convolutional neural networks arXiv Prepr 14045997
38. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: *advances in neural information processing systems* 25. *NIPS*
39. Kulkarni A, Ralph D, McFarlane D (2007) Value of RFID in remanufacturing. *Int J Serv Oper Informatics* 2:225–252. <https://doi.org/10.1504/IJSOI.2007.015326>
40. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521:436–444. <https://doi.org/10.1038/nature14539>
41. Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86:2278–2324. <https://doi.org/10.1109/5.726791>
42. Lee C-M, Woo W-S, Roh Y-H (2017) Remanufacturing: trends and issues. *Int J Precis Eng Manuf Technol* 4:113–125. <https://doi.org/10.1007/s40684-017-0015-0>
43. Lee D-H, Kang J-G, Xirouchakis P (2001) Disassembly planning and scheduling: review and further research. *Proc Inst Mech Eng Part B J Eng Manuf* 215:695–709. <https://doi.org/10.1243/0954405011518629>
44. Li J, Feng J, Kuo C-CJ (2018) Deep convolutional neural network for latent fingerprint enhancement. *Signal Process Image Commun* 60:52–63. <https://doi.org/10.1016/J.IMAGE.2017.08.010>
45. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft COCO: common objects in context. In: *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*. Springer, Cham, pp 740–755
46. Ramachandran M, Ragavendran U, Fegade V (2018) Selection of used piston for remanufacturing using fuzzy TOPSis optimisation. In: *Fuzzy systems and data mining IV : Proceedings of FSDM*. IOP Press, pp 61–67
47. Mairal J (2016) End-to-End Kernel Learning with Supervised Convolutional Kernel Networks ArXiv Prepr 160506265
48. Matsumoto M, Yang S, Martinsen K, Kainuma Y (2016) Trends and research challenges in remanufacturing. *Int J Precis Eng Manuf Technol* 3:129–142. <https://doi.org/10.1007/s40684-016-0016-4>
49. Neal RM (1992) Connectionist learning of belief networks. *Artif Intell* 56:71–113. [https://doi.org/10.1016/0004-3702\(92\)90065-6](https://doi.org/10.1016/0004-3702(92)90065-6)
50. Nwankpa C, Ijomah W, Gachagan A, Marshall S (2018) Activation Functions: Comparison of Trends in Practice and Research for Deep Learning arXiv Prepr 181103378
51. Paterson DAP, Ijomah WL, Windmill JFC (2017) End-of-life decision tool with emphasis on remanufacturing. *J Clean Prod* 148:653–664. <https://doi.org/10.1016/j.jclepro.2017.02.011>

52. Ponce J, Forsyth D (2012) Computer vision: a modern approach. **Pearson**
53. Popov BN (2015) Corrosion engineering: principles and solved problems. Elsevier Inc.
54. Ren R, Hung T, Tan KC (2018) A generic deep-learning-based approach for automated surface inspection. *IEEE Trans Cybern* 48:929–940. <https://doi.org/10.1109/TCYB.2017.2668395>
55. Rios A, Kavuluru R (2015) Convolutional neural networks for biomedical text classification. In: Proceedings of the 6th ACM conference on bioinformatics, computational biology and health informatics - BCB '15. ACM Press, New York, pp 258–267
56. Robotis A, Boyaci T, Verter V (2012) Investing in reusability of products of uncertain remanufacturing cost: the role of inspection capabilities. *Int J Prod Econ* 140:385–395. <https://doi.org/10.1016/j.ijpe.2012.04.017>
57. Shamma N, Jain V, Mishra A (2018) An analysis of convolutional neural networks for image classification. *Procedia Comput Sci* 132:377–384. <https://doi.org/10.1016/J.PROCS.2018.05.198>
58. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: ICLR
59. Steinhilper R (1998) Remanufacturing: the ultimate form of recycling. 1–24
60. Subramoniam R, Huisingh D, Chinnam RB (2009) Remanufacturing for the automotive aftermarket-strategic factors: literature review and future research needs. *J Clean Prod* 17:1163–1174. <https://doi.org/10.1016/j.jclepro.2009.03.004>
61. Sun M, Zhang X, Van Hamme H, Zheng TF (2016) Unseen noise estimation using separable deep auto encoder for speech enhancement. *IEEE/ACM Trans Audio, Speech, Lang Process* 24:93–104. <https://doi.org/10.1109/TASLP.2015.2498101>
62. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 1–9
63. USITC (2012) Remanufactured goods : an overview of the U . S . and Global Industries , Markets and Trade
64. Wang H, Bengio S (2002) THE MNIST DATABASE of handwritten upper-case letters. In: Training. <http://yann.lecun.com/exdb/mnist/index.html>. Accessed 26 Jul 2018
65. Weng R, Lu J, Tan Y-P, Zhou J (2016) Learning cascaded deep auto-encoder networks for face alignment. *IEEE Trans Multimed* 18:2066–2078. <https://doi.org/10.1109/TMM.2016.2591508>
66. WHO (2017) The top 10 causes of death in 2015. World Health Organization
67. Yazid AM, Rijal JK, Awaluddin MS, Sari E (2015) Pattern recognition on remanufacturing automotive component as support decision making using Mahalanobis-taguchi system. *Procedia CIRP* 26:258–263. <https://doi.org/10.1016/j.procir.2014.07.025>
68. Yosinski J, Clune J, Bengio Y, Lipson H (2014) How transferable are features in deep neural networks? In: advances in neural information processing systems (NIPS). Pp 3320–3328
69. Zhang Y, Zhou D, Jiang P, Zhang H (2011) The state-of-the-art surveys for application of metal magnetic memory testing in remanufacturing. *Adv Mater Res* 301–303:366–372. <https://doi.org/10.4028/www.scientific.net/AMR.301-303.366>
70. Zikopoulos C, Panagiotidou S, Nenes G (2010) The value of sampling inspection in a single-period remanufacturing system with stochastic returns yield. *IFIP Adv Inf Commun Technol* 338 AICT:128–135. doi: https://doi.org/10.1007/978-3-642-16358-6_17

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

Affiliations

Chigozie Nwankpa¹ · Solomon Eze¹ · Winifred Ijomah¹ · Anthony Gachagan² · Stephen Marshall²

✉ Chigozie Nwankpa
Chigozie.nwankpa@strath.ac.uk

¹ Design, Manufacturing and Engineering Management, University of Strathclyde, Glasgow G1 1XJ, UK

² Electronic and Electrical Engineering, Faculty of Engineering, University of Strathclyde, Glasgow G1 1XJ, UK