Deep Learning Based Vision Inspection System for Remanufacturing Application

Chigozie Nwankpa[1], Solomon Eze[1], Winifred Ijomah[1], Anthony Gachagan[2], Steven Marshal[2]

[1] Design, Manufacturing and Engineering Management, [2] Electronic and Electrical Engineering

Faculty of Engineering, University of Strathclyde Glasgow G1 1XJ.

Abstract

Deep Learning has emerged as a state-of-the-art learning technique across a wide range of applications, including image recognition, localisation, natural language processing, prediction and forecasting systems. With significant applicability, Deep Learning is continually seeking other new fronts of applications for these techniques. This research is the first to apply Deep Learning algorithm to inspection in remanufacturing. Inspection is a key process in remanufacturing, which is currently an expensive manual operation in the remanufacturing process that depends on human operator expertise, in most cases. This research further proposes an automation framework based on Deep Learning algorithm for automating this inspection process. The proposed technique offers the potential to eliminate human factors in inspection, save cost, increase throughput and improve precision.

This paper presents a novel vision-based inspection system on Deep Convolution Neural Network (DCNN) for three types of defects, namely pitting, surface abrasion and cracks by distinguishing between these surface defected parts. The materials used for this feasibility study were 100cm x 150cm mild steel plate material, purchased locally, and captured using a web webcam USB camera 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, especially for accuracy and speed.

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*

Corresponding Author: Chigozie.nwankpa@strath.ac.uk

1.0 Introduction

Remanufacturing has been identified as a potential technique of achieving sustainable consumption of raw materials, energy and environmentally safe production [1], [2]. It is an industrial process by which eligible used products and/or their sub-units known as cores are brought to a ‘as good as new’ functional condition with matching warranty [3], [4]. 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 [5].

As an industrial process, remanufacturing is made up of the following three independent operations [6]:

- Assembly
- Processing which includes processes such as cleaning, inspection, testing and other operations that can bring the used product to an ‘as new’ condition.
- Reassembly.

Remanufacturing is referred to as an ultimate form of recycling because it essentially utilises the remaining added value in products by re-using them in their original geometric form to yield products that are as good as new ones; while creating new business opportunities, jobs, and increasing product affordability.[7], [8]. 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 ([8], [9].

In the developed world, remanufacturing helps businesses to keep up with regulations aimed at controlling environmental pollution and wasteful consumption habits. In the European Union, two of such regulations include the End-of-life vehicle (ELV) and the waste electrical and electrical (WEEE) equipment legislations [10], [11]. 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 re-use 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 cost to bear in this arrangement. However, by implementing remanufacturing, manufacturers can potentially make profits while keeping with the requirements of these regulations.

Early studies valued remanufacturing in the United States of America and the United Kingdom at USD 53 billion and GBP 2.4 billion respectively, while recognising that it still has great potential to grow [12]. However, much of the anticipated growth is delayed by the very complicating nature of remanufacturing. Critical areas of complication are [9], [13]:

- Difficulty 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 disassembly, inspection and cleaning efficiently and without difficulty.
- Possible rejection of remanufactured products by customers

This study focuses on the aforementioned second class of complications and specifically, the inspection application.

1.1 Inspection in Remanufacturing

In remanufacturing, inspection is a key process used to guarantee quality and reliability [14]. Consequently, inspection cells are often integrated into functional remanufacturing workshop floor [15] to facilitate core classification and/or to determine their extent of deterioration and quality [16]. Inspection reduces the risk of 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. Besides, accurate inspection techniques have been found to increase the remanufacturer's profit, satisfy the customers [17] and mitigate risks associated with uncertainty in core supply and quality [18].

A minimum of two primary stages of inspection is usually carried out in a typical remanufacturing process: Core inspection and post disassembly inspection of components. The main objective of core inspection is to determine the economic and reliability feasibility of remanufacturing a core. For instance, this may be aimed at determining the core's design model, quality or identification of some indicators that point to its remanufacturability [19]. Post disassembly inspection on the hand is conducted on components or sub-assemblies after disassembly to determine whether they could be re-used or not.

Efficient Information management is vital to remanufacturing inspection; scholars [20], [21] 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.

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 [22]. Similarly, Zhang et al. [23] proposed a metal magnetic memory (MMM) inspection technique for predicting the residual life of structural cores. The authors demonstrate the potentials of MMM in detecting macro and micro-crack as well as in predicting the residual useful life (RUL) of a core, which is necessary to ensure that re-used cores do not have inherent faults that would cause premature product failure.

The requirements of existing inspection technologies potentially complicate the inspection process; making a new and more effective solution expedient. Hence, this study proposes a vision-based inspection system, which can contribute to cost-effective remanufacturing automation. The following sections introduce the novel system.

1.2 Proposed DL Based Vision Inspection System for Remanufacturing.

The DL based vision inspection system for remanufacturing is an inspection technique that explores the use of neural networks to achieve system inspection. These 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 models are generally referred to as the convolutional neural networks (CNNs). These architectures are inspired by the natural mechanism of visual perception of living creatures[24] and have been successfully applied in image recognition, object detection and localisation [25]–[27].

The CNN is a type of the 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 and followed by the fully connected layers and used mostly for image analysis[28].

The first CNN used for recognition task was named LeNet5, developed and used for digit recognition[29] and after then, other architectures of CNN have evolved. Other CNN based architectures for object recognition includes; AlexNet[30], VggNet[31], SegNet[32], ResNet[33], SeNet[34] model and many other models.

The CNN architecture used for the development of this inspection system is the 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[33]. The basic idea of the building block of the ResNet architecture is shown.

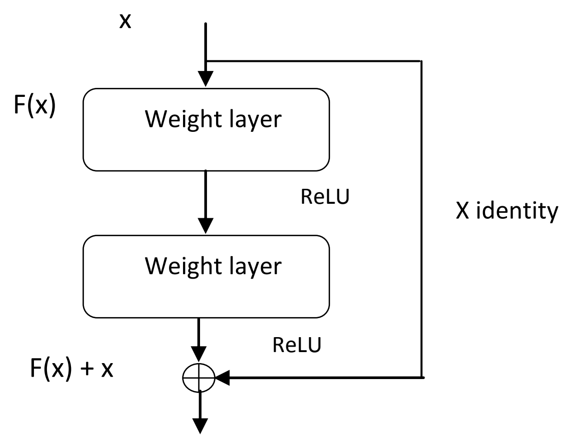

Figure 1 ResNet residual learning building block

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. Also, the deep residual networks proved that increased depth of the network provided higher accuracy[33].

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[33]. 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 improve the recognition accuracy.

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^ias weights for i filters, and filter bias b^i, activation function σ, a small image patch x_s and input image size of M x N, the convolution of filter i and input x_s is given by

$$f_{i\,s} = \sigma\left(W^i x_s + b^i\right) \qquad (1)$$

By sliding the patch of size $a\,x\,b$ across the image, and using a stride of size s, we obtain a convolution output given by [28]

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

where $\lceil . \rceil$ 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[35].

The pooling layers down-samples the spatial dimensions of any given input image. They are applied conventionally after the convolution layers to reduce the learned features dimensionality and avoid overfitting of the network. The pooling operations can either be average pooling, max-pooling or global average pooling, however, most DCNN architectures have the max-pooling used in their design and the max-pooling is given by [28]

$$Maxpool = \max(x_s) \quad (3)$$

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

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

The fully connected layers make up the last layers of most CNN architectures, used to compute class scores. However, the convolutional layers have also been used lately in the final layers of the network architectures, to learn the feature maps, end –to –end[36].

The early traditional vision-based inspection systems explored hand-crafted algorithms to extract the interest points in data such as edges [37], [38] and corner detectors[39], [40], scale-invariant feature transform[41], speeded-up robust features[42], the histogram of oriented gradients[43], and many other techniques. However, some other vital features considered when quantifying these component features of image data include shape, colour, and texture[44]. 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 compute the characteristics of given image sample manually.

2.0 Design and Implementation

The design and implementation of the deep learning-based vision inspection system is based on the real-time application of inspection. However, the training of the designed network to achieve the DL vision-based inspection system was a critical 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, number of epochs that represents the total number of iterations of the input data and other inherent considered factors.

This approach for solving the classification problem can be categorised into crucial process component stages, which include data collection where we recorded video data of samples, pre-processing is where the recorded video samples are converted to images and split the datasets into training and test examples, chose and design model where we select the appropriate computational model for the designed system and apply it to our problem, train model stage involves training the network to obtain the best weights for useful object classification, and lastly evaluate the model where we test the performance of the designed system. The process flow diagram is as shown.

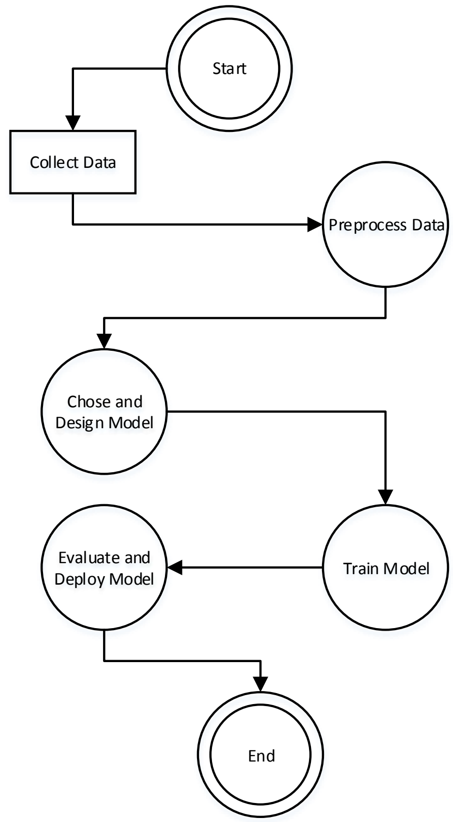

Figure 2 Model Development approach for the DL based vision inspection system.

Further details about the model development are outlined in the following sections to highlight some other specifics, to achieve the results presented in this paper.

I Collect Data

The data collection involved the use of the web webcam USB high definition camera of 0.3 megapixels, which was programmed to record three minutes of recorded video data of the samples in the lab. 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.

II Preprocessing Data

This stage involves capturing the required images needed and labelling each of the associated images. The categories include the eight identified types of fault that the algorithm will classify. The faults are pitting surface faults (PF), cracking surface faults (CF), rusting surfaces faults (RF), pitting and crack surface faults (PnC), rusting and crack surface faults (RnC), rusting and pitting surface faults (RnP) and pitting, crack and rough surface faults (PnRnC) alongside non-defected surfaces (Nodef). Also, resize the collected images to the desired size of the model was part of the pre-processing stage.

The output of the pre-processing stage produced images, which consist of some 3600 samples per class, and the eight classes produced the 28800 images used to train the network. The data was split to 80% training, and 20% of the available training samples was used to evaluate the deep learning-based vision inspection system, with. A total of 23040 image samples were used to train the network and 5760 image samples for the evaluation of the performance of the classifier.

II 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 chose the right 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 hence ResNet was the first deep CNN architecture that surpassed human-level performance on image recognition problem[33]. This model was modified to suit our data for training.

Another design consideration is the inference speed which is a concern, especially when there's need to deploy these models on lower performance systems. After deployment, the model inference was 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.

III Train Model

The training of the network is the most complicated stage of the deep learning-based vision inspection system. The training involves processes of trying 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 termed as hyperparameters, and they must be appropriately set for the adequate performance of the classifier.

The training set is used for training the ResNet model. In the training stage, the model learns how to recognise the respective categories of data available to our labelled data. In the event of a mistake, the classifier learns from its mistake and improves itself. The learning process involves the forward pass and backpropagation of the gradients obtained from the connected model weights. The test set was useful in making predictions about the input data by the classifier or for the evaluation of the developed algorithm for classification tasks. The primary issue avoided during the gathering of these data is a class imbalance where you have more examples of a type of fault than the others do and this biases the algorithms during training, thereby causing overfitting. This problem was avoided by using the same number of training examples.

IV Evaluate and Deploy Model

The evaluation stage involves presenting the split test set images to the developed classifier and make predictions on what the label of the image is. The results of the forecast are presented in tabular format for visualisation. The predicted models are compared against the ground-truth tags 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 visualisation to highlight 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.

2.1 Design and Implementation

The design of the DL vision inspection system is based on ResNet18 architecture. The 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 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[33].

2.2 Design Setup

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

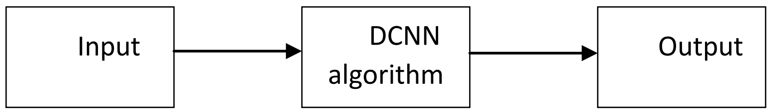

Figure 3 Block diagram of the system implementation

The input represents the set of videos captured from the connected cameras. The videos are converted to images and used in the model. 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 as the obtained deep learning vision-based inspection results. These outputs are displayed as labels and their prediction probability which are presented as results of the developed system. The entirety of these blocks makes up the developed deep learning-based vision inspection system for remanufacturing application.

The setup was achieved using a 2.7GHz, 8GB RAM computer with a GT740 graphics card and a web webcam USB camera of 0.3 megapixels. The material 100cm x 150cm mild steel plate with different defects was recorded as videos at about 0.5m distance from the camera. The camera was moved about to capture the data such that it reflects some of the critical issues face 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.

3.0 Design Performance Evaluation and Discussion

The ResNet architecture was chosen because of the accuracy of the architecture with Yosinki et al., 2014 outlining that if we have a small dataset, transfer learning is the best approach to best fine-tune the network for efficient learning[45]. The weights were frozen during the network training, achieved using transfer learning. The classification layers were also modified to suit the vision inspection dataset.

The hyperparameters of the systems include the learning rate, which was appropriately fixed at 0.001. A batch size of 20 was chosen carefully such that the model converges correctly during training as Krizhevsky, 2014 highlighted that huge batch sizes affects the quality of the final solution[46]. 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, was set to 4 as the model trains on vast parameters which would cause the training to last very long. A training and validation accuracies of 100% and 99.65% was obtained on the designed system after training of the model. These results are shown in the training progress response shown below.

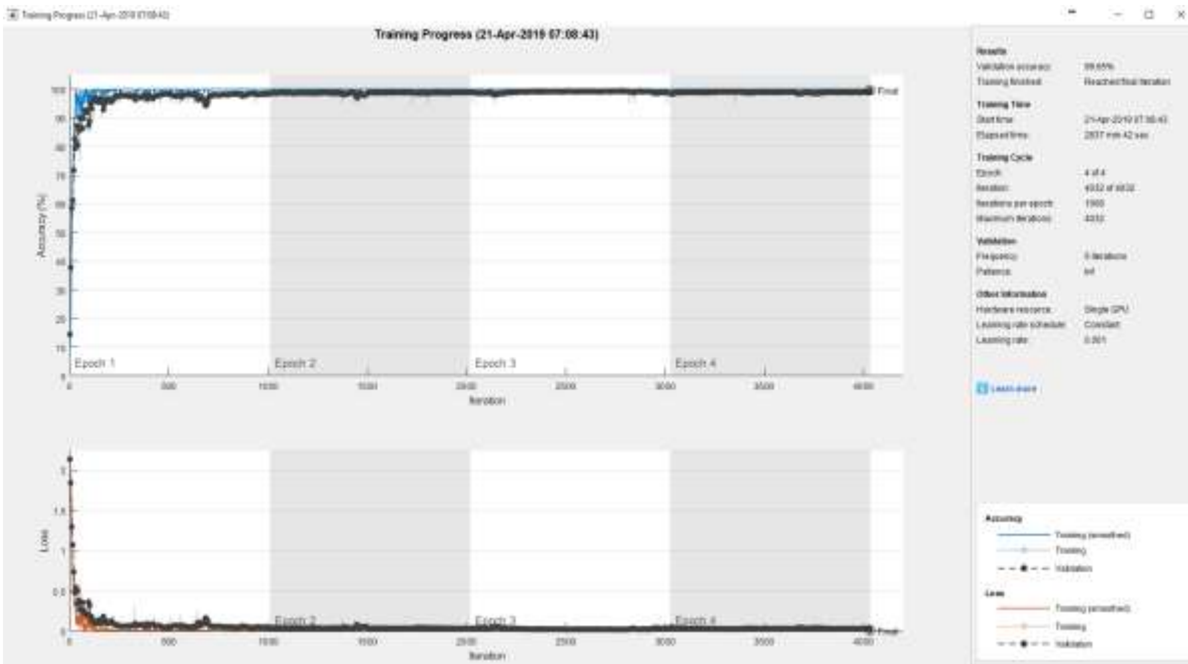

Figure 4 Training Progress Results

The evaluation of the designed system was performed at two stages, which include; using the validation data and using live camera feed.

The developed model produced an 8 x 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 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 that the system performance is of reasonably high accuracy.

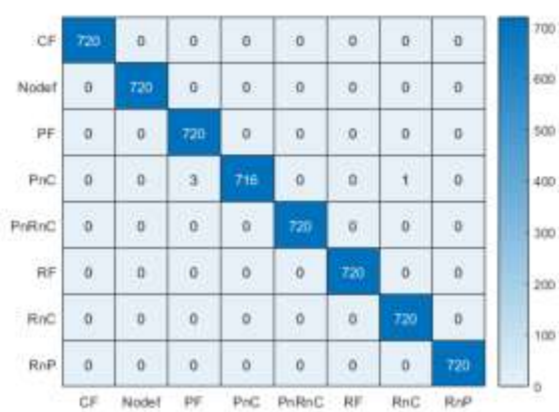

Figure 5 Confusion matrix visualisation of the evaluation results of the DL based vision system.

The second stage of the testing involved the use of the video feed, which is the USB camera which captures the surfaces for inspection. The system processes the captured picture frames from the live camera video stream. It performs the desired inspection, with the results displayed within one second, as shown in the Figure 6. An accuracy of 96 – 100% was achieved on the tested samples.

Figure 6 Video Inspection results showing video stream and inspection results

4.0 Conclusion

In this paper, a DL vision-based inspection system is presented. The method 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 show that the proposed method performed exceptionally well at approximately 100% accuracy on the validation data shown in the confusion matrix. This high performance was sustained when tested on a live video feed from a USB camera.

By adopting this model, we can achieve automated inspection in remanufacturing. This would enhance throughput, speed and reduce incidents of an industrial accident in remanufacturing especially when working around highly contaminated surfaces. This model aims to eliminate the sole dependence on expert judgment, which most remanufacturing companies depend on, to achieve accurate 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.

Acknowledgments

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

References

[1] W. L. Ijomah, S. J. Childe, G. P. Hammond, and C. A. McMahon, “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*, 2005, pp. 472–479.

[2] W. L. Ijomah, S. Childe, and C. McMahon, “Remanufacturing: A key strategy for sustainable development,” *3rd Int. Conf. Des. Manuf. Sustain. Dev.*, vol. 22, no. January, pp. 99–102, 2004.

[3] D. A. P. Paterson, W. L. Ijomah, and J. F. C. Windmill, “End-of-life decision tool with emphasis on remanufacturing,” *J. Clean. Prod.*, vol. 148, pp. 653–664, 2017.

[4] W. L. Ijomah, C. A. McMahon, G. P. Hammond, and S. T. Newman, “Development of design for remanufacturing guidelines to support sustainable manufacturing,” *Robot. Comput. Integr. Manuf.*, vol. 23, no. 6, pp. 712–719, 2007.

[5] USITC, “Remanufactured Goods : An Overview of the U . S . and Global Industries , Markets and Trade,” 2012.

[6] D.-H. Lee, J.-G. Kang, and P. Xirouchakis, “Disassembly planning and scheduling: review and further research,” *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, vol. 215, no. 5, pp. 695–709, Jan. 2001.

[7] R. Giutini and K. Gaudette, “Remanufacturing: The next great opportunity for boosting US productivity,” *Bus. Horiz.*, vol. 46, no. 6, pp. 41–48, Nov. 2003.

[8] R. Steinhilper, “Remanufacturing: the ultimate form of recycling.” Fraunhofer IRB Verlag, D-70569 Stuttgart, pp. 1–24, 1998.

[9] M. Matsumoto, S. Yang, K. Martinsen, and Y. Kainuma, “Trends and research challenges in remanufacturing,” *Int. J. Precis. Eng. Manuf. Technol.*, vol. 3, no. 1, pp. 129–142, Jan. 2016.

[10] F. Fakhredin, C. Bakker, J. Geraedts, and J. Huisman, “Five Perspectives on Design for End of Life: Highlights of a Literature Review.”

[11] WHO, “The top 10 causes of death in 2015,” World Health Organization, 2017.

[12] V. D. R. Guide Jr., “Production planning and control for remanufacturing: Industry practice and research needs,” *J. Oper. Manag.*, vol. 18, no. 4, pp. 467–483, 2000.

[13] R. Hammond, T. Amezquita, and B. Bras, “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.*, vol. 4, no. 1, pp. 27–46, 1998.

[14] S. T. M. Kin, S. K. Ong, and A. Y. C. Nee, “Remanufacturing Process Planning,” *Procedia CIRP*, vol. 15, pp. 189–194, Jan. 2014.

[15] H. K. Aksoy and S. M. Gupta, “Buffer allocation plan for a remanufacturing cell,” *Comput. Ind. Eng.*, vol. 48, no. 3, pp. 657–677, 2005.

[16] M. Errington, “Remanufacturing Inspection Models,” pp. 1–7, 2006.

[17] Q. Gu and T. Gao, “Management strategy of inspection error rate in reverse supply chain of remanufacturing,” *Jisuanji Jicheng Zhizao Xitong/Computer Integr. Manuf. Syst. CIMS*, vol. 22, no. 10, pp. 2469–2477, Oct. 2016.

[18] A. Robotis, T. Boyaci, and V. Verter, “Investing in reusability of products of uncertain remanufacturing cost: The role of inspection capabilities,” *Int. J. Prod. Econ.*, vol. 140, no. 1, pp. 385–395, Nov. 2012.

[19] C. Zikopoulos, S. Panagiotidou, and G. Nenes, “The value of sampling inspection in a single-period remanufacturing system with stochastic returns yield,” *IFIP Adv. Inf. Commun. Technol.*, vol. 338 AICT, pp. 128–135, 2010.

[20] A. Kulkarni, D. Ralph, and D. McFarlane, “Value of RFID in remanufacturing,” *Int. J. Serv. Oper. Informatics*, vol. 2, no. 3, pp. 225–252, Oct. 2007.

[21] L. F. Gildea and M. D. Adickes, “Front end shop floor control in a remanufacturing area,” *Tech. Pap. - Soc. Manuf. Eng. MS*, pp. 1–7, 1993.

[22] A. M. Yazid, J. K. Rijal, M. S. Awaluddin, and E. Sari, “Pattern recognition on remanufacturing automotive component as support decision making using Mahalanobis-Taguchi system,” *Procedia CIRP*, vol. 26, pp. 258–263, 2015.

[23] Y. Zhang, D. Zhou, P. Jiang, and H. Zhang, “The state-of-the-art surveys for application of metal magnetic memory testing in remanufacturing,” *Adv. Mater. Res.*, vol. 301–303, pp. 366–372, 2011.

[24] D. H. Hubel and T. N. Wiesel, “Receptive fields and functional architecture of monkey striate cortex.,” *J. Physiol.*, vol. 195, no. 1, pp. 215–43, Mar. 1968.

[25] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[26] K. He, X. Zhang, S. Ren, and J. Sun, “Delving Deep into Rectifiers: Surpassing Human-Level

Performance on ImageNet Classification," Feb. 2015.
[27] C. Szegedy *et al.*, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.
[28] R. Ren, T. Hung, and K. C. Tan, "A Generic Deep-Learning-Based Approach for Automated Surface Inspection," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 929–940, Mar. 2018.
[29] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, 2012.
[31] K. Simonyan and A. Zisserman, "VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION," 2015.
[32] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," Nov. 2015.
[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 770–778, 2016.
[34] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation Networks," *ArXiv Prepr. 1709.01507*, Sep. 2017.
[35] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning," *arXiv Prepr. 1811.03378*, Nov. 2018.
[36] J. Mairal, "End-to-End Kernel Learning with Supervised Convolutional Kernel Networks," *ArXiv Prepr. 1605.06265*, May 2016.
[37] A. K. Griffith, "Edge Detection in Simple Scenes Using a Priori Information," *IEEE Trans. Comput.*, vol. C-22, no. 4, pp. 371–381, Apr. 1973.
[38] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.
[39] H. Asada and M. Brady, "The Curvature Primal Sketch," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 1, pp. 2–14, Jan. 1986.
[40] F. Arrebola, P. Camacho, A. Bandera, and F. Sandoval, "Corner detection and curve representation by circular histograms of contour chain code," *Electron. Lett.*, vol. 35, no. 13, p. 1065, 1999.
[41] David G. Lowe, "Object Recognition from Local Scale-Invariant Features," in *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, 1999, p. 1150.
[42] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Comput. Vis. Image Underst.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.
[43] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 886–893.
[44] M. H. Bharati, J. J. Liu, and J. F. MacGregor, "Image texture analysis: methods and comparisons," *Chemom. Intell. Lab. Syst.*, vol. 72, no. 1, pp. 57–71, Jun. 2004.
[45] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 3320–3328.
[46] A. Krizhevsky, "One weird trick for parallelising convolutional neural networks," *arXiv Prepr. 1404.5997*, Apr. 2014.