Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 104 (2021) 827–832

www.elsevier.com/locate/procedia

54th CIRP Conference on Manufacturing Systems

Vision-Based Damage Localization Method for an Autonomous Robotic Laser Cladding Process

Habiba Zahir Imam[a], Yufan Zheng[a], Pablo Martinez[a], Rafiq Ahmad[a,*]

[a]*Laboratory of Intelligent Manufacturing, Design and Automation (LIMDA), Department of Mechanical Engineering, University of Alberta, 9211 116 Street NW, Edmonton, AB, T6G1H9, Canada*

* Corresponding author. *E-mail address:* rafiq.ahmad@ualberta.ca

Abstract

Currently, damage identification and localization in remanufacturing is a manual visual task. It is time-consuming, labour-intensive. and can result in an imprecise repair. To mitigate this, an automatic vision-based damage localization method is proposed in this paper that integrates a camera in a robotic laser cladding repair cell. Two case studies analyzing different configurations of Faster Region-based Convolutional neural networks (R-CNN) are performed. This research aims to select the most suitable configuration to localize the wear on damaged fixed bends. Images were collected for testing and training the R-CNN and the results of this study indicated a decreasing trend in training and validation losses and a mean average precision (mAP) of 88.7%.

Keywords: Remanufacturing; Deep neural networks; Damage localization; Robot laser cladding; Machine vision; Repair

1. Introduction

Laser Cladding (LC) or Laser-based Direct Metal Deposition (LMD) is an attractive additive manufacturing technique that has garnered considerable interest for applications in aerospace, oil and gas industry and mechanical engineering [1]. This well-established industrial process works by focusing a high-power laser beam to generate a molten pool on the substrate along with continuously directing material through a coaxial nozzle into that weld pool where it solidifies [2]. This layer-by-layer technique of manufacturing has the capability of increasing time and cost efficiency as compared to conventional technologies like casting, forging, and machining [3].

For several decades, LMD is being actively researched as an effective technology for repair and remanufacturing [4]. Repair or remanufacturing is credited for increasing the sustainability of the manufacturing sector by bringing damaged or worn metal parts back to like-new conditions. The process generally involves identifying and locating damages on a part's surface and then depositing material to restore the original geometry. Today, innovators in the field of repair are seeking to develop strategies to boost the level of automation for repair and maintenance and thereby boosting the flexibility of the process [5]. It has been shown that when integrated with a robotic manipulation system LMD shows an increased geometric flexibility, accessibility and saves production time [6].

In robotic laser cladding applications, inspection of the worn area is currently a manual process. The damage is visually localized by an operator who then uses a laser scanner to capture the surface geometry of the defect [7,8]. The information from this process is used to generate a repair strategy for the part. As the scale of the part increases, this procedure becomes more time-consuming, prone to error and labour-intensive.

Computer vision is an interdisciplinary field that seeks to understand, automate, and replace human visual tasks in any working environment. Moving towards an autonomous

10.1016/j.procir.2021.11.139

Robotic Laser Cladding Repair Cell (RLCRC), a significant amount of research is being done using artificial intelligence (AI), more specifically supervised learning methods such as deep learning to inspect the laser welding process [9,10]. Convolutional Neural Networks (CNN) are being trained to monitor and identify weld defects and melt pools during the laser cladding procedure [11,12]. Region-based Convolutional Neural Network (R-CNN) is a deep learning object detection approach where the R stands for regions of interest in an image. R-CNN first generates region proposals then uses CNN to extract features, locate and classify objects. Computer vision techniques paired with R-CNN's are being extensively adopted across many disciplines such as construction, transportation, materials science, geoscience and food production for automatic object detection and classification [13,14]. They have also made their way into manufacturing for damage detection and classification [15,16]. Additionally, intelligent vision-based practices are being implemented on shop floors for classifying and automating the repair inspection process [17].

Faster R-CNN has a notably speedier object detection time compared with previous image classification and object detection models [18]. It was developed to function closest to real-time reaching ten times the speed of Fast R-CNN [19]. You Only Look Once (YOLO) and Single Shot multi-box Detector (SSD) Mobilenet have a higher detection speed then Faster R-CNN's but a notably lower accuracy [20]

R-CNN's are deep and complex networks that require a significant amount of time and data to reach desirable results. Transfer learning is a promising learning framework that essentially transfers knowledge learnt in a previous task to a novel task; proven to save time and give effective results when data is scarce [21].

Based on the above literature review, it is evident that computer-vision based deep learning techniques have a lot of potential and have shown promising results across many disciplines. Despite that, remanufacturing remains to be heavily reliant on human intervention for damage detection and localization. There is immense scope for these intelligent strategies to be used to automate the damage localization process in a real RLCRC. This would be incumbent in achieving a fully autonomous repair system. With recent advances in computer vision and the availability of abundant data, it is economically worthwhile to explore the use of this technology in an RLCRC. This paper first proposes an integration of a vision sensor in a repair cell to record image data of damaged components. Then, two case studies are carried out utilizing two different datasets. These case studies perform analyses to compare the viability, accuracy and time efficiency of popular feature extractors for damage detection purposes. Finally, based on the results a suitable model configuration is selected and the results and evaluation provided.

2. Methodology

This study focuses on damage identification and localization on cylindrical fixed bends, more specifically the damage and the pad on fixed bends. These are mechanical parts used in the oil and gas industry. For worn fixed bends, it is essential to distinguish the location of the pad, as it is the area that incurs the most damage and must be repaired.

It is important to note here that this method must work hand in hand with a depth sensor to get an accurate volumetric representation of the damage. The results from this study can easily be extended to serve that purpose. However, that lies outside the scope of this paper.

2.1 Vision-based RLCRC

The robot arm used is Fanuc-R-1000iA/80F which is a high-speed handling robot for medium payloads and the camera is UVC-G3-Bullet/UVC-G3-AF. A schematic of the setup of the cell is demonstrated in Figure 1.

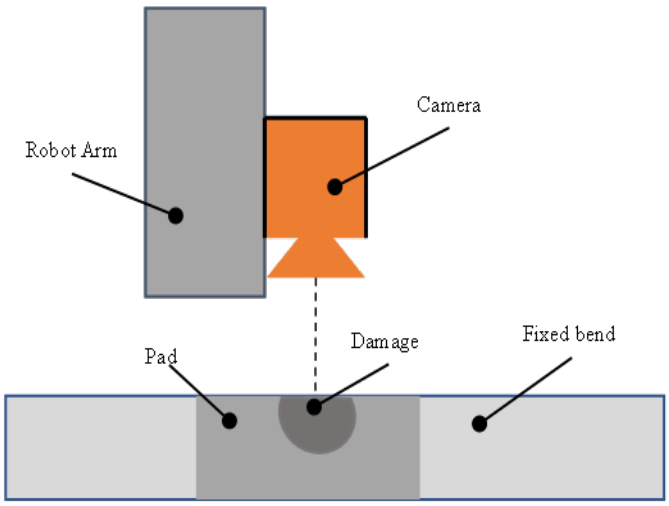

Fig. 1. Setup of the RLCRC

Configuration of the Object Detection Model

As established earlier, the aim of this paper is to select the best configuration of a model for damage detection that functions closest to real-time. Therefore, this study utilizes Faster R-CNN for damage detection and localization.

Faster R-CNN is an object detection architecture that comprises a feature extraction network, a region proposal (RPN) network and a region of interest (ROI) network. The pre-processed images first go through a pre-trained CNN (e.g., ResNet, inception etc.). Then the RPN generates possible regions of interest in the image. Finally, the ROI predicts the class label and the bounding box.

The open-source TensorFlow object detection API is used with TensorFlow version 1.5. Applying the concept of transfer learning, the network is initialized with weights that are pre-trained on the common objects in context (COCO) dataset and are present in the TensorFlow model zoo. The feature extractors analyzed in this study are Resnet50 and Inception V2. Table 1 highlights these architectures used in literature and their purpose.

Table 1. Different architectures used in literature for damage localization

Meta-architecture	Feature extractor	Objective	Paper
Faster R-CNN	ResNet50	Crack detection	[22]
Faster R-CNN	Inception-V2	Defect detection	[23]

3. Results and Discussion

3.1 Experimental Setup

Experiments are carried out on Google Colaboratory (RAM~ 12.6 GB, GPU: Tesla K80,12 GB, Disk~ 33GB).

Values for the hyperparameters in the feature extractor configuration pipeline are chosen by monitoring the progression of mean average precision (mAP) and training and validation losses. The parameters are adjusted iteratively to minimize losses and maximize mAP values while keeping the duration of training manageable with the available computational resources. The set of values found to be optimal are given in Table 2.

Table 2. Hyperparameters for training the model

Num of steps	Batch size	Learning rate	Score threshold	Momentum optimizer
20,000	12	0.001	0.2	0.9

3.2 Case Study 1

3.2.1 Dataset

To develop a database containing images of damaged fixed bends, 72 images (resolution: 1920 x 1080 pixels) of 8 different types of fixed bends are collected. R-CNN's require a massive amount of training data to generate a high-performing model. This can be a burdensome task as obtaining a large amount of data is expensive and often not readily accessible. To overcome this problem, data augmentation is a widely embraced practice. For this study, different types of geometric (horizontal flip and vertical flip) and photometric (grayscale, hue and exposure) augmentation techniques are applied to render the training model more robust and resilient to lighting and camera setting changes. Figure 2 shows sample images from the expanded dataset which is enlarged by augmentation from 72 to 221 images.

The extended dataset is annotated using labelling [24], a graphical image annotation tool. The two labels for classification are 'damage' and 'pad' on fixed bends. The dataset is then randomly split into 70%, 20% and 10% for training, validation and testing data, respectively.

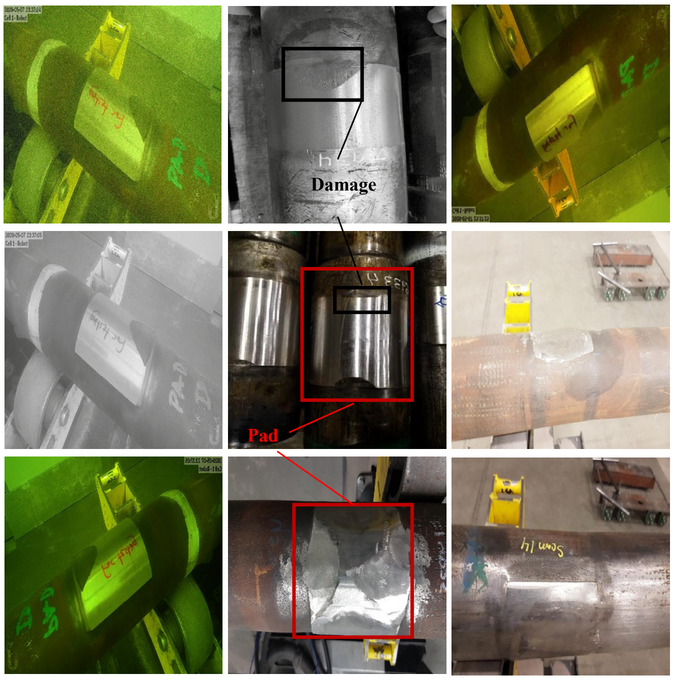

Fig 2. Sample augmented images from training dataset

3.2.2 Comparative Analysis and Results

This study primarily evaluates mAP as opposed to object proposal proxy metrics because it is a widely used metric for object detection [19].

Additionally, time taken for inference per image also plays an important role when implementing it in a real-world scenario. Thereby, two feature extractors are compared on their mAP and detection speed values and their results are tabulated in Table 3. As a reference, the publicly available mAP scores from the COCO dataset are also listed [25].

Table 3. Comparative analysis of the architectures trained with two labels

Architecture	'Fixed Bends' mAP	COCO mAP	Detection speed (ms/image)
ResNet50	**52.8%**	30%	1.48
Inception v2	49.1%	28%	**1**

Compared to COCO the results with the 'Fixed Bends' dataset are favorable, which is expected since COCO is a diverse dataset with 80 or more object categories. With only two categories ('pad' and 'damage'), higher mAP values should be achievable.

From Figure 2,it is apparent that the two labels have similar features and a constant overlap in the images. These factors are hypothesized to be a reason for creating bias and variance in the model, resulting in the relatively low mAP scores. To investigate this hypothesis, the models are trained again but this time with one label ('pad'). For the reason established earlier in section 2, the most vital information when repairing fixed bends is the location of the pad. From Table 4, it is evident that the model performance has drastically improved.

Table 4. Comparative analysis of the architectures trained with one label

Architecture	'Fixed Bends' mAP	COCO mAP	Detection speed (ms/image)
ResNet50	**70.4%**	30%	1.48
Inception v2	60.8%	28%	**1**

Training and validation losses for ResNet50 with one label are presented in Figure 3. Inference is performed on both model configurations and the bounding box predictions are demonstrated in Figure 4

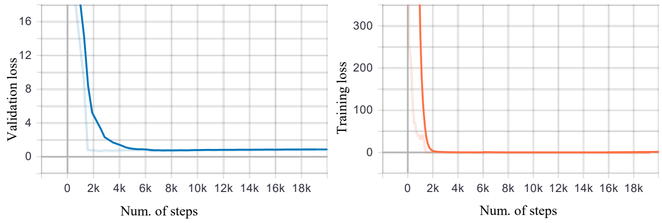

Fig. 3. Training and validation losses vs number of steps

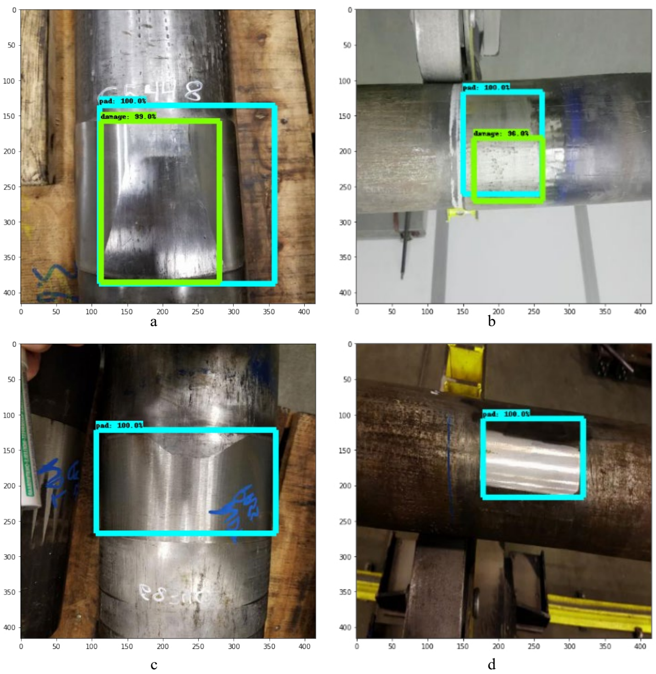

Fig. 4. Testing dataset with bounding box output. Trained with (a,b) two labels 'pad' and 'damage'; (c,d) one label 'pad'

3.3 Case Study 2

3.3.1 Dataset

For autonomous damage detection, the position of the camera in the RLCRC will remain unchanged, meaning the images from the camera of the workstation will always be taken from the same setting. It was hypothesized that training the model with images of different fixed bends taken from the same position, will further improve the accuracy of the deep learning model. To this aim, a new dataset was formed containing images of similar orientation as those that the model will be expecting to see while it carries out damage detection.

A new dataset was formed that comprised 437 original images (resolution: 1920 x 1080 pixels) of four different fixed bends. Similar to the first dataset, the images were augmented to expand the dataset to 1049 images (see Figure 5). The images were annotated using labellmg [46], this time for one label 'pad' because of the higher performance achieved using one label as observed in section 3.2.2. The dataset was then randomly split into 70%, 20% and 10% for training, validation and testing data, respectively.

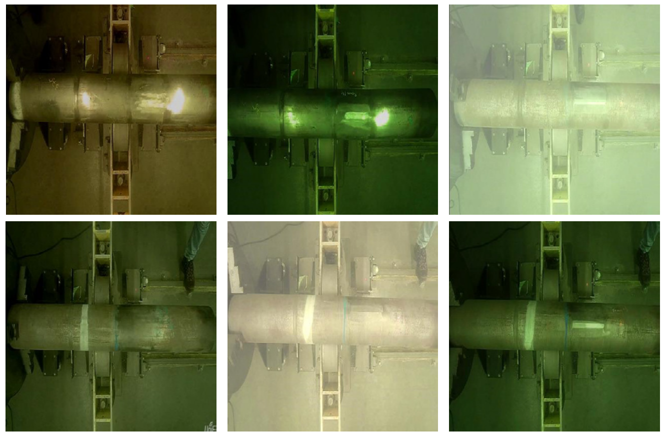

Fig. 5. Sample augmented images from the new training dataset containing only images with the same camera setting

3.3.2 Comparative Analysis and Results

Table 5 outlines the new results obtained from the model trained in case study 2, with the superior results formatted in bold. The new model significantly outperforms the model trained in case study 1. The highest achieved mAP score with the ResNet50 architecture increases from 70.4 in case study 1 to 88.7 in case study 2. This practically represents an improvement in accuracy of 26%.

Figure 6 illustrates the resulting metric plots obtained from the ResNet50 model. The training and validation losses, as shown in Figure 6(c, d), both decrease to a point of stability, which implies there is no overfitting. This model is assessed over several *IoU* metrics (*IoU*=0.50:0.05:0.95), which means the model has to be performing well at every *IoU* threshold for it to achieve a high *mAP* score. Figure 6(b) shows the *mAP* value at 0.50 *IoU*, reaching 100%.

Table 5: Comparative analysis of the architectures trained on the ne w dataset with one label

Architecture	'Fixed Bends' mAP	COCO mAP	Detection speed (ms/image)
ResNet50	**88.7%**	30%	1.48
Inception v2	79.4%	28%	**1**

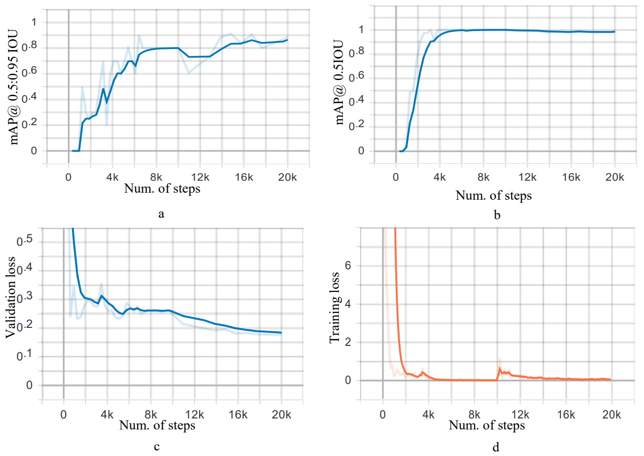

Fig. 6. Resulting metric plots showing (a) mAP @ 0.5:0.95 IOU & (b) mAP @ 0.5 IOU; (c) validation loss and (d) training loss

3.4 Discussion and Limitation

For case study 1, the model is being trained to detect pads and damages on fixed bends. Both classes have similar features and always overlap, which adds another level of complexity for the model. To mitigate this, the model is trained with only one label 'pad'. Doing so increases the mAP by 33% from 52.8 to 70.4 but this comes at the cost of not detecting the damage and pad separately. Since detection of pads is more vital for fixed bends, this is considered as the optimal choice.

Figure 4 shows that the prediction of pad locations is more concise for one label as opposed to two. For example, Figure 4a shows how when working with two labels, the bounding box of the pad does not cover the full width of the pad as opposed to Figure 4c where the pad width is entirely covered. It appears that with two labels, the 'pad' and 'damage' bounding boxes tend to share some of their boundary lines which leads to the localization of the pad being inaccurate. This could be due to the RPN and ROI regression being faulty. Analyzing alternate ways of object detection that support better decision overlap could improve the detection results.

The aim of this study is to develop an intelligent vision system that can identify and localize a damaged area. This localization process is performed using a fixed camera orientation, which means that the view of the camera remains unchanged throughout the process and between different parts. Therefore, it is more important to have a specialized model to localize the *'pad'* surface with a higher accuracy for the setup proposed rather than a robust model with a much lower accuracy. For this purpose, a second case study is performed with a new dataset consisting of images taken from the same camera orientation for one label *'pad'*. The results from this second model indicate a *mAP* score of 88.7, which is an increase of 26% compared to results from case study 1 for one label. Overall, an improvement of 35.9 in *mAP* is achieved by moving from a more diverse dataset trained with two labels to a less diverse dataset with one label. This represents a relative increase in accuracy of 68%.

The first case study is carried out on a relatively small dataset of 72 original images of eight fixed bends, whereas the second case study has 437 original images of four fixed bends. Results from the second case study are more favorable as the objective is to obtain a well-trained, more specialized model for detecting damages in a specific environment. A much larger dataset would enable the R-CNN to more accurately understand features of the damage and the pad and yield a more robust and higher performing model. Furthermore, GPU limitation on Google Colab restricted testing for a greater number of steps and experimenting with different hyperparameter settings to fine-tune the model and enhance performance. Access to more computational power would also make it possible to train and compare deeper architectures like ResNet101 for which training is more computationally expensive.

4. Conclusion and Future Work

Damage identification and localization in remanufacturing is a manual visual task. It can be time-consuming, tedious and prone to error. With recent advances in computer vision, computational power and access to a large amount of data it is now worthwhile to explore the use of this technology in remanufacturing. This paper proposes a machine learning-based method for automatic visual detection and localization of damages in a robotic laser cladding repair cell process. To accomplish this, two configurations of Faster R-CNN combining transfer learning are employed. Two case studies are performed on different datasets, case study 1 with a more diverse set of images and case study 2 with more similar images. The comparative analyses of their performance are also carried out. For case study 1, the model is trained with one label and two labels. The highest *mAP* score obtained while training for two labels is 52.8 using ResNet50 as the feature extractor. With the same feature extractor, the *mAP* score increases to 70.4 when training for only one label. For case study 2, the model is trained with one label only. The resulting model outperforms those from case study 1, reaching a maximum *mAP* of 88.7. The best model configuration in all cases is found to be Faster R-CNN with ResNet50 as the feature extractor. This model achieves a detection speed of 1.48 ms, rendering it potentially viable for real-time application. Promising results from this study demonstrate the potential of vision-based R-CNN technology in the field of repair and remanufacturing.

It is important to note that the scope of this paper is to find the best model for damage detection of fixed bends. The method presented in this paper will be extended to work with depth sensors and obtain volumetric information in the future that will be required to repair the part.

Acknowledgements

We express our gratitude to Group Six Technologies Inc. for their assistance and technical support. We also express our appreciation to Mario A. Soriano from the Laboratory of Intelligent Manufacturing, Design and Automation (LIMDA) for sharing his wisdom during the research. The authors acknowledge the Natural Sciences and Engineering Research Council (NSERC), Grant Nos. (NSERC RGPIN-2017-04516 and NSERC CRDPJ 537378-18) for funding this project.

References

[1] C. Leyens and E. Beyer, "Innovations in laser cladding and direct laser metal deposition," in *Laser Surface Engineering: Processes and Applications*, Elsevier Inc., 2015, pp. 181–192.

[2] M. Gharbi *et al.*, "Influence of a pulsed laser regime on surface finish induced by the direct metal deposition process on a Ti64 alloy," *J. Mater. Process. Technol.*, vol. 214, no. 2, pp. 485–495, 2014, doi: 10.1016/j.jmatprotec.2013.10.004.

[3] E. Brandl, A. Schoberth, and C. Leyens, "Morphology, microstructure, and hardness of titanium (Ti-6Al-4V) blocks deposited by wire-feed additive layer manufacturing (ALM)," *Mater. Sci. Eng. A*, vol. 532, pp. 295–307, 2012, doi: 10.1016/j.msea.2011.10.095.

[4] W. W. Liu, Z. J. Tang, X. Y. Liu, H. J. Wang, and H. C. Zhang, "A Review on In-situ Monitoring and Adaptive Control Technology for Laser Cladding Remanufacturing," *Procedia CIRP*, vol. 61, pp. 235–240, 2017, doi: 10.1016/j.procir.2016.11.217.

[5] E. Uhlmann, F. Heitmüller, M. Manthei, and S. Reinkober, "Applicability of industrial robots for machining and repair processes," *Procedia CIRP*, vol. 11, pp. 234–238, 2013, doi: 10.1016/j.procir.2013.07.042.

[6] Y. Ding, R. Dwivedi, and R. Kovacevic, "Process planning for 8-axis robotized laser-based direct metal deposition system: A case on building revolved part," *Robot. Comput. Integr. Manuf.*, vol. 44, pp. 67–76, 2017, doi: 10.1016/j.rcim.2016.08.008.

[7] H. Zheng, M. Cong, H. Dong, Y. Liu, and D. Liu, "CAD-based automatic path generation and optimization for laser cladding robot in additive manufacturing," *Int. J. Adv. Manuf. Technol.*, vol. 92, no. 9–12, pp. 3605–3614, 2017, doi: 10.1007/s00170-017-0384-0.

[8] H. Z. Imam, Y. Zheng, and R. Ahmad, "An efficient tool-path planning approach for repair of cylindrical components via laser cladding," *J. Remanufacturing*, vol. 10, no. 3, pp. 1–10, Nov. 2020, doi: 10.1007/s13243-020-00096-6.

[9] W. Cai, J. Z. Wang, P. Jiang, L. C. Cao, G. Y. Mi, and Q. Zhou, "Application of sensing techniques and artificial intelligence-based methods to laser welding real-time monitoring: A critical review of recent literature," *J. Manuf. Syst.*, vol. 57, no. May, pp. 1–18, 2020, doi: 10.1016/j.jmsy.2020.07.021.

[10] P. Pant and D. Chatterjee, "Prediction of clad characteristics using ANN and combined PSO-ANN algorithms in laser metal deposition process," *Surfaces and Interfaces*, vol. 21, no. July, p. 100699, 2020, doi: 10.1016/j.surfin.2020.100699.

[11] C. Xia *et al.*, "A review on wire arc additive manufacturing: Monitoring, control and a framework of automated system," *J. Manuf. Syst.*, vol. 57, no. August, pp. 31–45, 2020, doi: 10.1016/j.jmsy.2020.08.008.

[12] Y. Zhang, G. S. Hong, D. Ye, K. Zhu, and J. Y. H. Fuh, "Extraction and evaluation of melt pool, plume and spatter information for powder-bed fusion AM process monitoring," *Mater. Des.*, vol. 156, pp. 458–469, 2018, doi: 10.1016/j.matdes.2018.07.002.

[13] P. Martinez, M. Al-Hussein, and R. Ahmad, "Intelligent vision-based online inspection system of screw-fastening operations in light-gauge steel frame manufacturing," *Int. J. Adv. Manuf. Technol.*, vol. 109, no. 3–4, pp. 645–657, 2020, doi: 10.1007/s00170-020-05695-y.

[14] A. Reyes-Yanes, P. Martinez, and R. Ahmad, "Real-time growth rate and fresh weight estimation for little gem romaine lettuce in aquaponic grow beds," *Comput. Electron. Agric.*, vol. 179, 2020, doi: https://doi.org/10.1016/j.compag.2020.105827.

[15] Y. Zheng, H. Mamledesai, H. Imam, and R. Ahmad, "Deep Learning-based Automatic Damage Recognition and Spatial Localization for Remanufacturing/Repair," vol. 18, no. 6, pp. 381–385, 2021, doi: 10.14733/cadconfP.2020.381-385 Introduction:

[16] Y. J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyüköztürk, "Autonomous Structural Visual Inspection Using Region-Based Deep Learning for Detecting Multiple Damage Types," *Comput. Civ. Infrastruct. Eng.*, vol. 33, no. 9, pp. 731–747, 2018, doi: 10.1111/mice.12334.

[17] C. Nwankpa, S. Eze, W. Ijomah, A. Gachagan, and S. Marshall, "Achieving remanufacturing inspection using deep learning," *J. Remanufacturing*, vol. 10, no. 3, 2020, doi: 10.1007/s13243-020-00093-9.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 580–587, 2014, doi: 10.1109/CVPR.2014.81.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.

[20] Á. Arcos-García, J. A. Álvarez-García, and L. M. Soria-Morillo, "Evaluation of deep neural networks for traffic sign detection systems," *Neurocomputing*, vol. 316, pp. 332–344, 2018, doi: 10.1016/j.neucom.2018.08.009.

[21] S. Panigrahi, A. Nanda, and T. Swarnkar, "A Survey on Transfer Learning," *Smart Innov. Syst. Technol.*, vol. 194, no. 10, pp. 781–789, 2021, doi: 10.1007/978-981-15-5971-6_83.

[22] D. Kang, S. S. Benipal, D. L. Gopal, and Y. J. Cha, "Hybrid pixel-level concrete crack segmentation and quantification across complex backgrounds using deep learning," *Autom. Constr.*, vol. 118, no. May, 2020, doi: 10.1016/j.autcon.2020.103291.

[23] Z. Guo *et al.*, "Defect detection of nuclear fuel assembly based on deep neural network," *Ann. Nucl. Energy*, vol. 137, 2020, doi: 10.1016/j.anucene.2019.107078.

[24] Tzutalin, "LabelImg," *Git code*, 2015. [Online]. Available: https://github.com/tzutalin/labelImg.

[25] T. Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014, vol. 8693 LNCS, no. PART 5, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.