Automated Industrial Load Station Inspection

Yasir Ijaz
School of Computing, Eng & Intel. Sys
Ulster University
Londonderry, UK
Email: ijaz-y@ulster.ac.uk

Sonya Coleman
School of Computing, Eng & Intel. Sys
Ulster University
Londonderry, UK
Email: sa.coleman@ulster.ac.uk

Dermot Kerr
School of Computing, Eng & Intel. Sys
Ulster University
Londonderry, UK
Email: d.kerr@ulster.ac.uk

Nazmul Siddique
School of Computing, Eng & Intel. Sys
Ulster University
Londonderry, UK
Email: nh.siddique@ulster.ac.uk

Cormac McAteer
Seagate Technology
Springtown Industrial Estate
Londonderry, UK
Email: cormac.p.mcateer@seagate.com

Khoi Nguyen
Seagate Technology
Springtown Industrial Estate
Londonderry, UK
Email: khoi.k.nguyen@seagate.com

***Abstract*—The manufacturing sector heavily relies on fast and accurate material inspection to improve product quality and productivity. Computer vision approaches based on deep learning provide superior performance for automatic material inspection over traditional methods. However, collecting a large amount of data to train a deep learning model is a challenging and expensive task, especially in an industrial environment. To address this challenge, we propose a synthetic augmentation approach, in which source data is prepared with a hybrid technique, combining synthetic data and real data to train a deep learning model, with performance evaluated on the target data. The applied approach consists of three key steps and performance evaluation indicates the ability to learn the domain-variant features and enhance the model's generalisation ability. For inspection purposes, two well-known models, YOLOv5 and YOLOv8, were employed utilizing the optimizers SGD and AdamW, respectively, which provided significant test accuracy. Building on the results, the applied approach may be recommended for similar inspection use cases in smart industry applications.**

***Index Terms*—Computer vision, Synthetic Data, Transfer learning, Material inspection**

I. Introduction

In recent years, industries have rapidly adopted new technology, enhancing productivity and safety while reducing labour costs and environmental impact [1]. Industry 4.0 promotes automatic material inspection to improve product quality. According to the World Economic Forum, companies could increase operational efficiency by 20-30% from automation in industrial sectors like manufacturing and transportation energy [2]. However, automation in industrial systems faces various challenges and complex processes such as operating inspection activities. Therefore, designing an automatic inspection system in an industrial setting is a demanding task requiring a high upfront investment for long-term benefits. Deep learning is a cutting-edge technology that has proven ability to solve various problems in smart manufacturing, processing large volumes of data and extracting valuable information. However, these models assume a vast amount of data for training and testing which is a challenging task, especially for industrial applications [3]. This study addresses the issue of the wafer handling process in smart manufacturing where there is a need to train an automatic guided vehicle (AGV) to transfer semiconductor wafers from a production tool to a load station (LS). This task requires an automatic inspection system, and before loading wafers onto the LS there is a need to inspect the condition of the LS, to determine whether it is ready to receive the wafers. Currently, the process of inspecting a LS is performed manually by humans, which presents various challenges. The manual process requires professional workers, has a potential risk of human injuries, and is time-consuming and therefore resource-intensive. However, with advancements in technology, automatic operations have become the preferred choice for high precision and consistency in repetitive and demanding tasks. In this case, the AGV needs to determine the LS condition before receiving the cassette that holds the wafers. For this purpose, a computer vision system needs

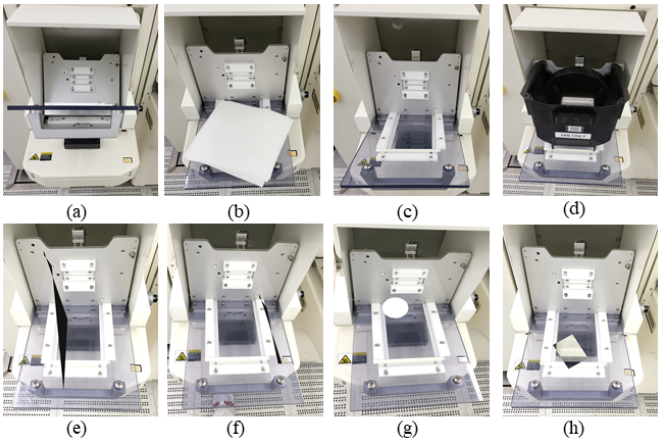


Fig. 1: Schematic illustrating real abnormal samples: (a) incorrect position, (b) tissue on LS indicating obstruction, (c) angle variation, (d) cassette on LS; and synthetic abnormal LS samples (e-h) integrated with synthetic obstructions of varying sizes and shapes.

to analyse if the LS has been opened correctly and is not obstructed. This implies that the station is completely open and ready for wafer insertion, whereas the inspection process cannot complete if the LS is in an inaccurate position or is obstructed, as illustrated in Figure 1(a-d). This study proposes a solution that uses a deep learning model with a hybrid approach to prepare the source domain data, including using only normal real images to learn the global knowledge for both the normal and abnormal states of the LS. Utilising this method reduces the data collection effort required, which is highly beneficial as collecting, annotating, and training with large datasets is time-consuming and demands computational power [4].

Recently, several computer vision models based on deep learning have been designed and implemented in smart manufacturing settings to inspect materials, resulting in a consistent progression towards intelligent manufacturing, which holds great potential for the future [5, 6, 7]. Material inspection based on deep learning can be divided into three main steps [8]: 1) Deep backbone architectures like VGG [9], Inception [10], ResNet [11], and Darknet [12] to extract features from the image, 2) Region selection strategies such as Faster R-CNN [13], YOLOv5 [14], and Single Shot Multibox Detector (SSD) [15], and 3) Learning strategies encompassing various techniques, including data augmentation [16], cascaded learning [17], transfer learning [18], and non-maximum suppression (NMS) [19]. There are two types of object detectors: a network that completes the detection process in two stages such as RCNN [20], Faster RCNN [13], and DetectoRS [21], and networks that complete the detection process in one stage such as YOLOv5 [14] and YOLOv8 [22]. YOLOv5 and YOLOv8 provide various weights trained on the COCO dataset [23], such as nano (n), small (s), medium(m), large(l), and extra large(x). Each weight offers different choices to accommodate different degrees of available resources and accuracy requirements. Chen et al. [24] proposed an approach using Faster RCNN to train the industrial robot UR5 for object localisation and detection among 50 classes. They employed a VGG backbone architecture for feature extraction, and the Caffe framework [25] was used to train the model. Similarly, another study introduced a deep learning-based approach to detect surface morphology based on YOLOv7 [26]. Their method utilised a small dataset for transfer learning to detect the surface morphology in direct energy decomposition. Zhu et al,. also proposed a study for automatic assembly quality inspection, in which synthetic data were generated and transfer learning was applied to retrain the weights using Faster RCNN [27]. Nikolenko et al, [28] also summarised the different approaches to generating synthetic data. A study proposed integration of real and synthetic data for assembly quality inspection, the synthetic data was prepared with a CAD model and then used with transfer learning and Faster RCNN [27].

The primary contributions of this paper are as follows:

- A key contribution is the ability to automatically determine if the LS is obstructed using real data and synthetic data.
- Reduce the additional annotation effort by using the same annotations for created abnormal samples that were previously used for the normal class.

II. Proposed Framework

The proposed approach consists of three steps to enhance the model generalisation: created synthetic obstructions, apply geometric data augmentation, and perform transfer learning as summarised in Fig 2. We use advanced single-stage object detectors previously trained on large datasets and perform transfer learning. The state-of-the-art YOLOv5 and YOLOv8 models are selected for training and the validation and testing results are based on the original data.

A. Dataset Preparation

The dataset was prepared in seven different ways, labeled D1 to D7, as can be seen in Table I. This enabled us to evaluate the model performance in various context to ensure the model generalisation ability for real industrial scenarios.

There were two types of LS images: normal images indicating that the LS is ready to perform the assembly operation, with the LS is in the right position, and there are no obstructions present; and abnormal images with various abnormalities, indicating that the assembly operation cannot be performed in this state. There are three real-world scenarios available for abnormal samples: class one (C1) consists of a cassette on the LS, class two (C2) consists of tissues on the LS, and class three (C3) where the LS is not in an accurate position. Some samples are shown in Figure 1 (a-d).

1) Geometric Augmentation: The process began with D1, which used only normal LS samples for model training. D2 was prepared to expand the D1 after applying geometric data augmentation. Various geometric transformation were performed including flipping, rotation, exposure correction, and hue adjustment applied to modify the colour tune of the image. These transformations increase the diversity of source domain data to improve the model's resilience capability. The subsequent configuration progressively incorporated abnormal samples, D3 to D5 were prepared with normal and random abnormal samples after applying geometric augmentations.

TABLE I: Dataset Preparation

Dataset	Train Data	Valid/Test Data
D1	Normal	Original Data
D2	Augmented Normal	Original Data
D3	Augmented Normal + C1	Original Data
D4	Augmented Normal + C1 + C2	Original Data
D5	Augmented Normal + C1 + C2 + C3	Original Data
D6	Hybrid data (Normal + Synthetic data)	Original Data
D7	**Augmented (Hybrid data)**	Original Data

2) Synthetic Obstruction Generation: D6 was introduced with another approach by using hybrid data to combine normal and synthetic abnormal samples. Generating synthetic data is an important approach to address the limitations of insufficient data quantities in deep learning models, which

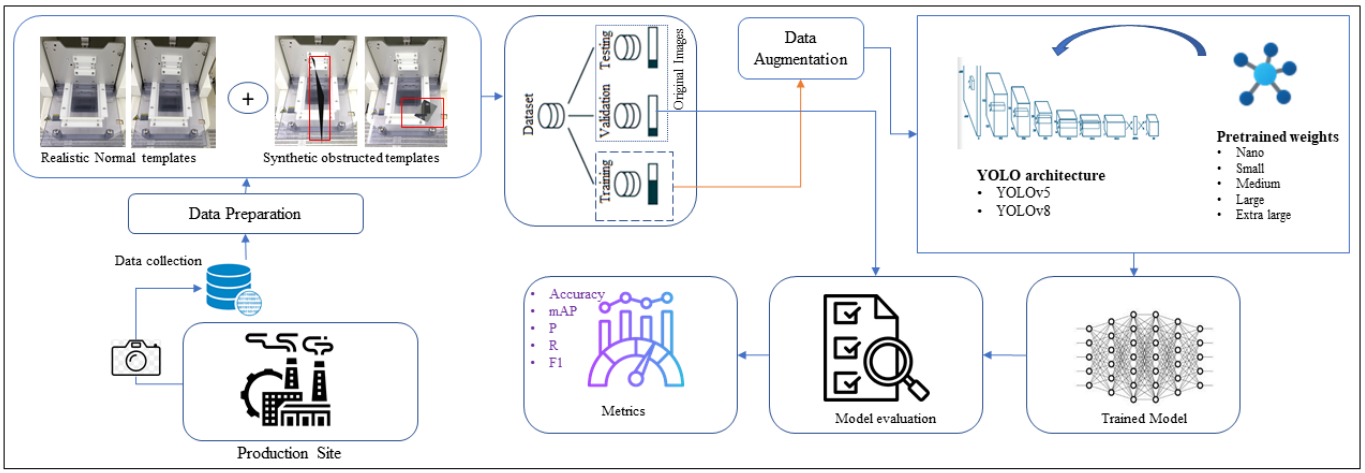


Fig. 2: Schematics diagram of the framework, integrating synthetic data creation, geometric data augmentation and transfer learning.

helps to enhance the variability of the source domain data. To overcome a limitation of model generalisation ability, we introduced synthetic obstructions on the surface of the normal LS images and prepared D6. This increased the diversity of data by incorporating abnormalities of various shapes, such as rectangles, circles, and polygons of different dimensions. The normal LS images have these synthetic obstructions superimposed on their surface. The obstructions were created on only one surface of the LS as shown in Figure 1(e-h), leaving the surrounding background untouched. We used pixel coordinates x and y from the annotated file to achieve this. To determine the exact location of the LS, we obtained the LS coordinates from an annotation text file (YOLO format) corresponding to each image. To train the model for global-level features, we added the obstruction classes, and adjusted the contrast and brightness for variable environments, as detailed in [29]. Thus abnormal samples are generated effectively using Equation 1.

$$\mathrm{O}_i(x,y) = \alpha \times (\mathrm{I}_i(x,y) + \beta) + \gamma \times S(x,y) \qquad (1)$$

where I_i is the i-th input image, O_i is the i-th output image, α represents the contrast, β represents brightness adjustment, and γ controls the impact of added shapes. $S(x,y)$ represents the shapes that were added to the image at pixel coordinates (x,y). To annotate the generated abnormal samples, we considered the annotated files of the normal class and changed the class label to abnormal, reducing the additional annotation effort.

Finally, D7 was prepared, yielding the best performance, combining the normal LS samples and hybrid augmented data. After generating the synthetic data, geometric data augmentation techniques were applied to D6 based on geometric transformations to prepare the D7. For model validation and testing, we considered real images captured with 2D camera in the manufacturing environment to ensure the model performance and generalisation ability. The datasets were split for training, validation, and testing in proportion of 70%, 20%, and 10%, respectively.

TABLE II: Selected hyperparametric structure to train YOLOv8 and YOLOv5

Hyperparameter	YOLOv8	YOLOv5
Input Size	640 × 640	640 × 640
Optimizer	AdamW	SGD
Optimizer Momentum	0.937	0.85
Batch Size	16	16
Training Epochs	150	150

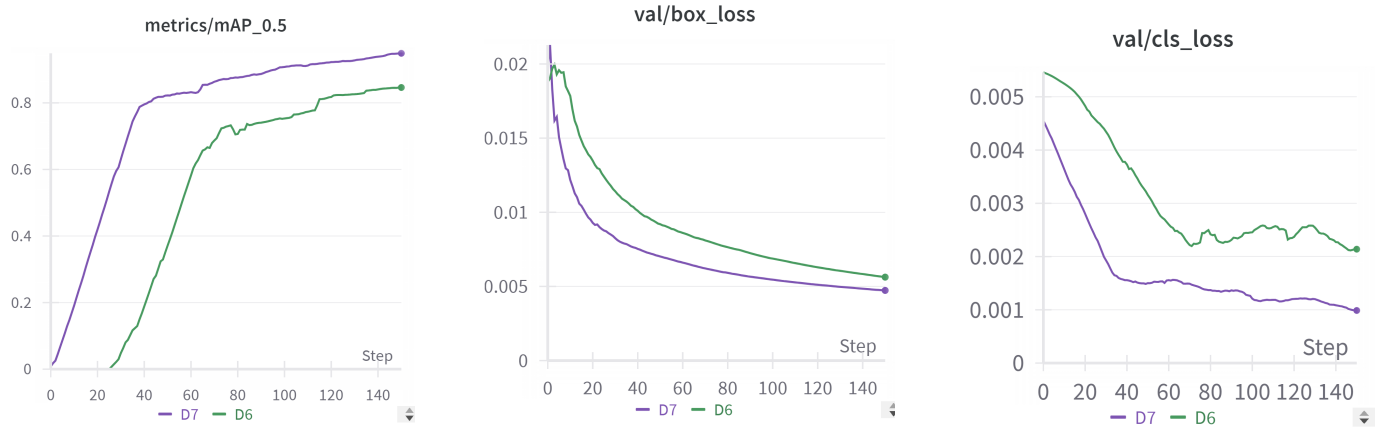
III. Model Implementation

Regarding YOLOv5 and YOLOv8, the SGD and AdamW optimisers respectively were observed to give optimal results alongside the corresponding hyperparameters. During the experimental phase, a standardised set of hyperparameters was applied to refine these models. In Table II the hyperparameters for both YOLOv5 and YOLOv8, which provided the optimal performance, are provided. Therefore, subsequent experiments were conducted using these selected parameters and performed model training across all prepared datasets.

A. Training and Validation

To determine the optimal parameters, YOLOv8 model tuning was performed with medium-size trained weights YOLOv8m, using 150 epochs with the available optimisers and parameters. For regularisation, an early stopping mechanism with patience 20 and a weight decay of 0.0005 were utilised. By utilising the patience value 20, the models learn early stopping if no further improvement is found in the last 20 epochs, hence each model completed their learning ability at different numbers of epochs.

The intersection over union measures the overlap between the predicted bounding box and the ground truth bounding box in object detection process. Precision, P, is the ratio



(a) (b) (c)

Fig. 3: The impact of geometric data augmentation in D7 over D6 is demonstrated by (a) the improved mAP, (b) reduced validation box loss, and (c) decreased class loss.

of the true prediction to the total positive predictions, used to measure positive prediction for normal LS state, whereas Recall, R, is the ratio of the true positive to the actual positive, used to measure the model's ability to identify the abnormal LS samples. Various forms of losses were also considered to assess the model performance, including box loss (box_loss), and classification loss (cls_loss). The box_loss parameter indicates the model's ability to accurately identify the centre of an object and assess the extent to which the predicted bounding box encompasses it. On the other hand, cls_loss parameters assess the model's performance in segmenting the object and correctly predicting its class based on the given sample. Average precision is used to analyse the model learning ability to identify and localise the LS, mean average precision *(mAP)* was considered to analyse the model learning ability. The models trained with D1 and D2 resulted in good *mAP*, up to 99%, with minimum losses for both training and validation using normal samples. However, the trained models with D1 and D2 learned the features for normal LS samples only, resulting in failure for abnormal samples. As we increase the number of abnormal classes in D3 to D5 and train the model with these datasets, the model learning ability improves for the abnormal state. However, after the evaluation we observed that the model still does not generalise well.

The model generalisation ability enhanced for both normal and abnormal classes when we trained with D6. Building on the results with D6, we further improve the model learning ability by training the model with D7, as seen in Figure 3 which indicates the improvement in mAP with decreases in box loss and class loss. The overall model learning ability enhanced when we trained the model on D7, as can be seen in Figure 4.

B. Testing and Evaluation Metrics

After training the model, our prime concern was to consider the model performance using real industrial data. The trained models using all prepared datasets were evaluated on

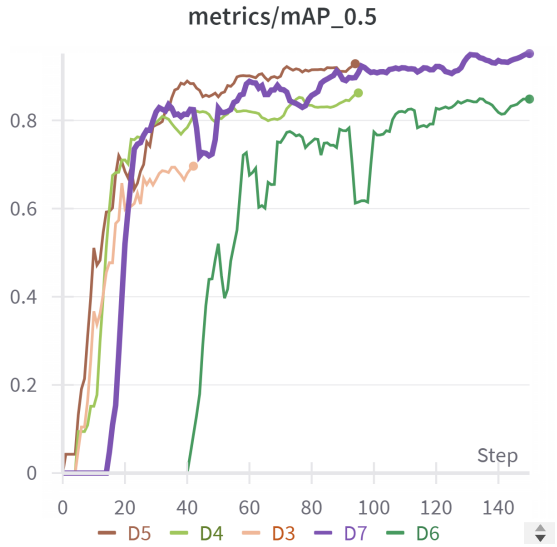


Fig. 4: The graph illustrates the improvement in model training performance for normal and abnormal LS features, with the highest *mAP* observed when training on dataset D7.

the original industrial data. In order to clearly differentiate between normal and obstructed LS instances, the confidence level had to be tuned, and finally, it was adjusted to 0.9. Furthermore, a threshold was implemented to ensure that only one detection per image was considered to minimise potential ambiguities. To maintain the accuracy and reliability of the evaluation process, in occurrences in which the LS was nearly closed as shown in Fig 1(c), the image was labeled as 'no detection'.

To verify the performance of the models, they were tested using industrial data and the performance was measured using accuracy, P, R, and F1_score according to Eq 2-5.

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{2}$$

$$P = \begin{cases} \frac{\text{TP}}{\text{TP+FP}}, & \text{if TP} + \text{FP} \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

$$R = \begin{cases} \frac{\text{TP}}{\text{TP+FN}}, & \text{if TP} + \text{FN} \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

$$\text{F1\_score} = \begin{cases} \frac{2\times(P\times R)}{P+R}, & \text{if } P + R \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

Here, TP is the actual number of obstructed LS images that the model correctly predicts as obstructed. FP is the number of normal states of the LS, but the model incorrectly predicts them as obstructed. TN is the number of normal images of the LS, and the model correctly predicts them as normal. FN is the number of obstructed LS, but the model incorrectly predicts them as normal. The models were tested on unseen image samples to ensure their performance.

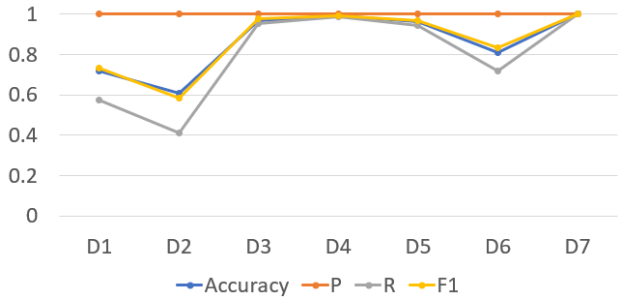
TABLE III: Testing results for YOLOv8m using the prepared datasets, highlighting the effectiveness of D7 in improving the model inspection performance.

Dataset	**Accuracy (%)**	***P* (%)**	R (%)	**F1_Score (%)**
D1	71.76	100	57.71	73.18
D2	60.88	100	41.41	58.57
D3	97.06	100	95.59	97.75
D4	99.12	100	98.68	99.33
D5	96.18	100	94.27	97.05
D6	81.18	100	71.81	83.59
D7	100	100	100	100

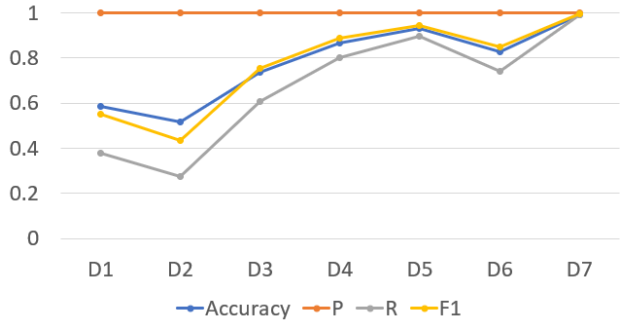
TABLE IV: Testing results of YOLOv5m using the prepared datasets, highlighting the effectiveness of D7 in improving the model inspection performance.

Dataset	**Accuracy (%)**	***P* (%)**	***R* (%)**	**F1_Score (%)**
D1	58.53	100	37.89	54.95
D2	51.76	100	27.75	43.45
D3	73.82	100	60.79	75.62
D4	86.76	100	80.18	88.99
D5	93.24	100	89.87	94.66
D6	82.65	100	74.01	85.06
D7	99.41	100	99.12	99.56

The models trained with D1 to D5 provide good test accuracy, however, they detect only those samples containing the abnormalities which were considered for the training purpose, indicating poor generalisation. With these datasets the model learning ability for normal samples was robust, however, the model was less generalisable for abnormal samples. The model trained with D6 demonstrates improved performance for both the normal and abnormal classes. When we applied geometric augmentation on D6 and prepared dataset D7, model generalisation was greatly improved; the testing accuracy improved to 100% for YOLOv8 and 99.41% for YOLOv5 based on the target samples. This is illustrated in Figure 5a and Figure 5b.



(a)



(b)

Fig. 5: The outcomes demonstrates the enhanced performance of the hybrid technique using D7. The results show improved metrics across different datasets with YOLOv8 (a) and YOLOv5 (b) emphasizing the effectiveness of the hybrid approach.

TABLE V: Performance comparison of YOLOv8 and YOLOv5 with other weights, model trained on D7.

Model	Weight	Accuracy (%)	P (%)	R (%)	F1 (%)
YOLOv8	YOLOv8n	95.88	94.19	100	97.01
YOLOv8	YOLOv8s	97.06	96.97	98.68	97.82
YOLOv8	YOLOv8l	97.94	97.01	100	98.48
YOLOv8	YOLOv5n	98.24	100.00	97.36	98.66
YOLOv8	YOLOv5s	98.27	97.42	100	98.70
YOLOv8	YOLOv5m	98.24	97.42	100	98.70
YOLOv8	YOLOv5l	97.65	97.00	99.56	98.26
YOLOv8	YOLOv5x	98.24	97.42	100	98.70
YOLOv5	YOLOv5n	98.53	98.26	99.56	98.90
YOLOv5	YOLOv5s	99.71	99.56	100	99.78
YOLOv5	YOLOv5m	99.41	100	99.12	99.56
YOLOv5	YOLOv5x	100	100	100	100

YOLOv8, using this approach achieved 100% *R* indicating that all abnormal images were detected correctly.

The results presented in Table III and Table IV indicate that our applied method effectively achieved significant testing performance using the target industrial data, and enhanced the model generalisation. The model demonstrated good generalisation ability by effectively handling both normal and abnormal samples that were completely unseen during the model training. These results indicate the model's capacity to handle the unseen samples. In addition, to test the performance with D7 we considered other weights for YOLOv5 and YOLOv8, and the performances are provided in Table V. These results

indicate that the detection results are relatively better, and the model has generalised to detect the normal and abnormal states for the LS.

IV. Conclusion

This paper presents an approach for load station inspection using real and synthetic data, which includes three steps: synthetic obstruction created on the surface of the normal samples to make them abnormal, data augmentation to increase the diversity of data and model implementation. To identify the best performing approach, we prepared the data in different ways and applied transfer learning using YOLOv5 and YOLOv8. The key finding is that by using this approach with transfer learning, the model achieves optimal performance with 100% accuracy, signifying the generalisation and effectiveness for load station inspection. The strength of this work is to enhance the model generalisation ability with a limited dataset to learn the image features. Additionally, the results indicate the model performance is consistent despite the variation of the obstruction types and can be applied in different industrial settings.

Acknowledgments

This research is supported by UKRI Strength in Places Fund Project (81801): Smart Nano-Manufacturing Corridor.

References

[1] Y. Lu and T. Ng, "Technological innovation and productivity gains," *Journal of Productivity Analysis*, vol. 49, no. 3, pp. 201–210, 2018.

[2] W. Lehmacher, F. Betti, P. Beecher, C. Grotemeier, and M. Lorenzen, "Impact of the fourth industrial revolution on supply chains," 2017.

[3] J. Smith and E. Johnson, "Deep learning applications in smart manufacturing: Opportunities and challenges," *Journal of Intelligent Manufacturing*, vol. 35, no. 2, pp. 123–145, 2024.

[4] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, pp. 261–318, 2020.

[5] M. Bugatti and B. M. Colosimo, "Towards real-time in-situ monitoring of hot-spot defects in l-pbf: a new classification-based method for fast video-imaging data analysis," *Journal of Intelligent Manufacturing*, vol. 33, no. 1, pp. 293–309, 2022.

[6] X. Qi, G. Chen, Y. Li, X. Cheng, and C. Li, "Applying neural-network-based machine learning to additive manufacturing: current applications, challenges, and future perspectives," *Engineering*, vol. 5, no. 4, pp. 721–729, 2019.

[7] S. K. Everton, M. Hirsch, P. Stravroulakis, R. K. Leach, and A. T. Clare, "Review of in-situ process monitoring and in-situ metrology for metal additive manufacturing," *Materials & Design*, vol. 95, pp. 431–445, 2016.

[8] H. M. Ahmad and A. Rahimi, "Deep learning methods for object detection in smart manufacturing: A survey," *Journal of Manufacturing Systems*, vol. 64, pp. 181–196, 2022.

[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition(CVPR)*, 2015, pp. 1–9.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[14] G. Jocher and A. Chaurasia, "ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation," Nov. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.7347926

[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[16] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.

[17] Viola and Snow, "Detecting pedestrians using patterns of motion and appearance," in *Proceedings ninth IEEE international conference on computer vision*. IEEE, 2003, pp. 734–741.

[18] L. Torrey and J. Shavlik, "Transfer learning," *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, vol. 1, pp. 242–264, 2010.

[19] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-nms–improving object detection with one line of code," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5561–5569.

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer vision and Pattern Recognition (CVPR)*, 2014, pp. 580–587.

[21] S. Qiao, L.-C. Chen, A. Yuille, and B. Xiao, "Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution," *In proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[22] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics yolov8," 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[23] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," *In proceedings of the European conference on computer vision*, pp. 740–755, 2014.

[24] X. Chen and J. Guhl, "Industrial robot control with object recognition based on deep learning," *Procedia CIRP*, vol. 76, pp. 149–154, 2018.

[25] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[26] X. Zhu, F. Jiang, C. Guo, D. Xu, Z. Wang, and G. Jiang, "Surface morphology inspection for directed energy deposition using small dataset with transfer learning," *Journal of Manufacturing Processes*, vol. 93, pp. 101–115, 2023.

[27] X. Zhu, P. Mårtensson, L. Hanson, M. Björkman, and A. Maki, "Automated assembly quality inspection by deep learning with 2d and 3d synthetic cad data," *Journal of Intelligent Manufacturing*, pp. 1–16, 2024.

[28] S. I. Nikolenko, *Synthetic data for deep learning*. Springer, 2021, vol. 174.

[29] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.