

# Low-Cost Real-Time Automated Optical Inspection Using Deep Learning and Attention Map

Yu Shih, Chien-Chih Kuo and Ching-Hung Lee\*

Department of Electrical and Computer Engineering, National Yang Ming Chiao Tung University, Hsinchu, 300, Taiwan

\*Corresponding Author: Ching-Hung Lee. Email: chleenctu@nctu.edu.tw

Received: 23 January 2022; Accepted: 19 April 2022

**Abstract:** The recent trends in Industry 4.0 and Internet of Things have encouraged many factory managers to improve inspection processes to achieve automation and high detection rates. However, the corresponding cost results of sample tests are still used for quality control. A low-cost automated optical inspection system that can be integrated with production lines to fully inspect products without adjustments is introduced herein. The corresponding mechanism design enables each product to maintain a fixed position and orientation during inspection to accelerate the inspection process. The proposed system combines image recognition and deep learning to measure the dimensions of the thread and identify its defects within 20 s, which is lower than the production-line productivity per 30 s. In addition, the system is designed to be used for monitoring production lines and equipment status. The dimensional tolerance of the proposed system reaches 0.012 mm, and a 100% accuracy is achieved in terms of the defect resolution. In addition, an attention-based visualization approach is utilized to verify the rationale for the use of the convolutional neural network model and identify the location of thread defects.

**Keywords:** Automated optical inspection; deep learning; real-time inspection; attention

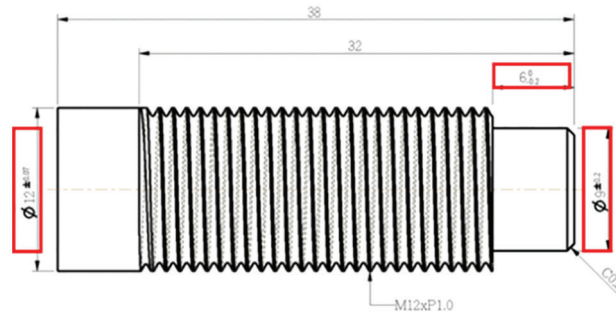
## 1 Introduction

As the basis of Industry 4.0, automated inspection is widely performed in various manufacturing applications to ensure consistent product quality [1–7]. However, manual and sampling inspections remain prevalent in quality control for reducing manufacturing costs; additionally, monitoring systems to reduce losses caused by machine tool abnormalities are insufficient. In previous studies, the importance of system integration and costs are not considered in the system design. Consequently, a low-cost automated optical inspection (AOI) system that integrates a production line to inspect products completely without adjustment is required. In addition, the results of automated full inspection or product monitoring can be used to directly diagnose the tool wear status of machine tools directly [8–10]. This provides a low-cost monitoring function for ensuring product quality without installing expensive monitoring equipment. This study focuses on the development of an AOI system using deep learning for



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

the real-time inspection of screws. An automated mechanism was designed, and inspection image recognition was used to identify the size and thread defects in screws (as shown in Figs. 1 and 2), thereby ensuring consistent product quality.



**Figure 1:** Technical drawing of a screw



**Figure 2:** Detection results of (a) normal (qualified) and (b) defective thread

In automation, the mode of transportation affects the speed and efficiency of inspection. Currently, production line transportation is typically performed using robot manipulators [11–14]. Robot manipulators can improve the adjustability of the process; however, calibration for different tasks is time consuming, particularly in collaborative robotic systems. Another typical transportation method involves automatic guided vehicles (AGVs) [15–17]. AGVs with suitable path-planning algorithms used in intelligent workshops render material transportation more efficient. However, AGVs are more suitable for transporting bulk or large materials. Moreover, companies with low capital cannot afford the high costs of robots and land. Because the conveyor is suitable for transporting unconfined materials [18,19] and does not require calibration, it is typically used in workshops. To establish an inspection system rapidly, a conveyor was adopted in this study. In addition, image histogram equalization and Canny edge detection are typically used to improve the detection accuracy of image recognition systems for determining the dimensions of an object [20–22]. Recently, detection consistency has been improved using deep learning methods, particularly convolutional neural networks (CNNs) [23–28]. Among CNNs, VGG16 is trained using one million images, which contain 1000 categories that encompass almost all objects in daily life [29]. The complex structure and significant amount of data of VGG16 endow it with high feature extraction ability. In this study, VGG16 with transfer learning was used to establish the proposed defect detection system. To verify the detection results, two visualization methods with attention mapping, gradient-weighted class activation mapping (Grad-CAM) [30,31], and Grad-CAM++ [32,33] were used to

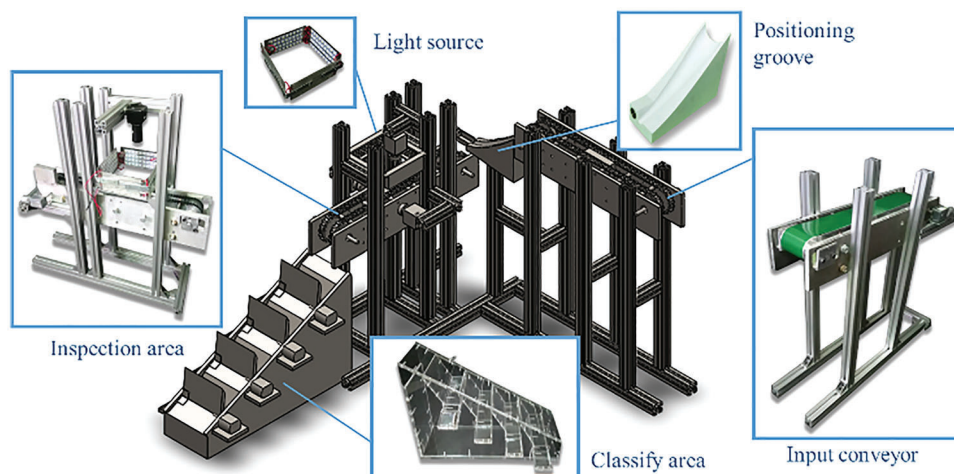
generate attention maps. Unlike previous visualization methods [34–36], Grad-CAM does not require model structure changes to provide effective explanations. The areas of interest of the model are denoted in the attention maps with weights. Grad-CAM++ improves the weight  $w_k^c$ , which considers the pixel location. In this study, actual differences caused by pixel location abnormalities are compared. Attention maps are used to denote the locations of defects that assist in the reverification process.

System integration is a complex process for achieving intelligent manufacturing, particularly for workshops with low capital. To reduce production costs, the equipment used often spans multiple generations and is sourced from different suppliers, rendering system integration difficult. Therefore, an add-on quality control system is introduced to provide the maximum production benefits. The design of the mechanism allows the AOI system to be connected to the production line without additional calibration. In addition, the transfer learning method for the defect detection model can reduce the number of computer calculations required. The full evaluation of the dimension and the defect detection results can be recorded via real-time full inspection; as such, condition monitoring without requiring numerous sensors in the equipment is achieved. Finally, the results of the attention map verify the rationale for using the CNN model and denoting the location of thread defects, which improve the credibility of the model.

The remainder of this paper is organized as follows: Section 2 introduces the system mechanism and hardware specifications. The established deep learning defect detection method is introduced in Section 3. Section 4 presents the corresponding experimental results and discussion. Finally, the conclusions are presented in Section 5.

## 2 Mechanism Design

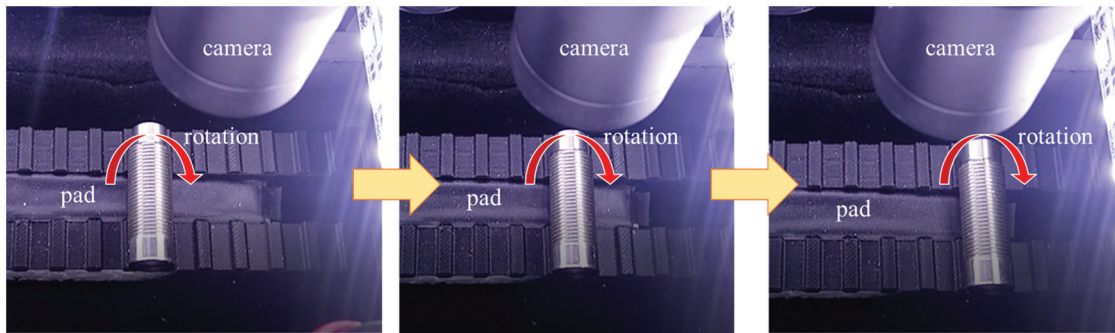
Fig. 3 shows the mechanism of the automated real-time inspection system. As shown in Fig. 3, the proposed system includes an input conveyor, position grooves, an inspection area, and a classification region. To achieve real-time inspection, two conveyors are used to integrate the proposed system with the machining process. Screw positioning, image capture, and recognition are achieved for each screw during movement. The proposed system can be placed directly at the end of the production line without adjustment.



**Figure 3:** Mechanism of automated real-time inspection

The feeding conveyor transports the screws from the machining process to the positioning mechanism. The feeding conveyor belt, which is flat and high speed, is used to move the workpieces. To ensure stability

during inspection, a double-sided toothed belt is used in the detection conveyor to fix the screw in place, thereby preventing slippage between the belt and pulley. In addition, a pad is placed in the middle of the detection conveyor to roll screws over this area; thus, the upper camera can capture multiple images with different thread angles to ensure the integrity of detection. The mechanism for rolling the screw as it passes through the upper camera is illustrated in Fig. 4.



**Figure 4:** Mechanism of rolling the screw as it passes through upper camera

In this study, a mechanism that does not require additional power is designed to fix the direction and position of the screw. The mechanism comprises a landslide and an aluminum block. The shape of the landslide is similar to that of a funnel, which fixes the screw to the positioning block in a particular direction. Subsequently, the weight of the aluminum block enables the screw to be turned over and placed securely on the detection conveyor when it enters the aluminum block. After the screw departs from the aluminum block, it automatically returns to its original position. The positioning process can be completed without additional power sources or sensors.

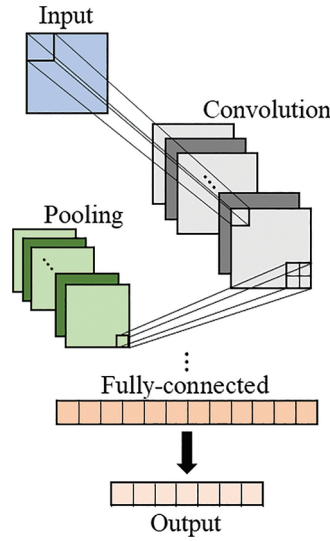
The light source directly affects the image quality. In this study, four LED strips were arranged in a square to achieve an effect similar to that of a ring light. In the classification area, four servomotors were used to control the four barriers. The classification area was classified into one area for qualified screws and three areas for unqualified screws, namely those with a large diameter, a small diameter, or thread defects. After an object is inspected, a barrier opens based on its classification, and the other barriers close, forming a slope that enables the screws to enter the appropriate area. The system is integrated with Arduino Uno programmed using Python and C#. The Arduino Uno is matched with L298N to operate the motor at the required speed. To facilitate operation, the camera, motor, and light source are integrated with the C# user interface through the Arduino Uno. In addition, the interface communicates with Python to update the dimension measurements and defect detection results. The proposed system combines image recognition and deep learning to measure the dimensions of the thread and identify its defects within 20 s, which is lower than the production-line productivity per 30 s. Thus, the AOI system can achieve real-time quality control.

### 3 Defect Detection Using Deep Learning with Attention Map

The proposed defect-detection method using a CNN with an attention map is introduced in this section. Grad-CAM++ was adopted to create the defect region for monitoring and reverification. First, we used discontinuous edge images to determine the dimensions of the recognized objects. After image preprocessing, the Sobel operator and Canny edge detection were used to identify boundaries by adjusting the threshold.

### 3.1 CNN

Neural networks can mitigate fluctuations in inspection results caused by the manual inspection of abundant data; meanwhile, extensive calculations, which are performed via mathematical or computational models that mimic the structure of biological neural networks, are performed in machine learning and cognitive science to approximate neurological functions. Neural networks are adaptive systems, and the most typically used type of deep learning method in image processing is CNNs [23–28]. CNNs are advantageous owing to their capability to automatically extract features from images. CNNs comprise three layers, i.e., convolutional, pooling, and fully connected layers, as illustrated in the architecture shown in Fig. 5.



**Figure 5:** Convolutional neural network structure

#### 3.1.1 Convolutional Layer

The convolutional layer contains a filter matrix for calculating the output neurons via local input weights and the connected region. In a grayscale image, the convolutional operation is expressed as follows:

$$(I \otimes K)_{ij} = \sum_{m=1}^m \sum_{n=1}^n K_{m,n} I_{i+m,j+n}, \quad (1)$$

where  $I \otimes K$  denotes the convolutional operations of the images and kernels;  $m$  and  $n$  denote the pixels in the  $m$ -th row and  $n$ -th column, respectively. Subsequently, after convolution, a nonlinear activation function such as a sigmoid or rectified linear unit is used to determine the output (feature map), as follows:

$$C_{m,n}^{p,q} = ReLU\left(\sum_{m=1}^m \sum_{n=1}^n I_{m-u,n-v} K_{u,v}^{p,q} + b^{p,q}\right) \quad (2)$$

#### 3.1.2 Pooling Layer

The pooling layer is used to reduce the size of the input image. In the pooling process, the convolutional feature matrix is partitioned into regions, and the maximum or average values of each region are obtained. The pooling operation is expressed as

$$P_{m,n}^{p,q} = \max(C_{m,n}^{p,q}) \quad (3)$$

Extracting representative features from the pooling layer significantly reduces the number of parameters.

### 3.1.3 Fully Connected Layer

The fully connected layer calculates the classification score and determines the final category. The operation of the  $j$ -th neuron in the fully connected layer  $L$  is expressed as

$$y_j^L = f\left(\sum_{i=1}^i x_i^{L-1} w_{ij}^L + b_j^L\right), \quad (4)$$

where  $x_i^{L-1}$  denotes the input of the  $j$ -th neuron,  $w_{ij}^L$  the weight of the input  $x_i^{L-1}$ ,  $b_j^L$  the bias,  $f$  the activation function, and  $y_j^L$  the corresponding output.

### 3.2 Attention-based CNN

In numerous deep-learning models, multilayer networks are used to automatically select features to achieve high accuracy. In most CNNs, global average pooling (GAP) is used in class activation mapping (CAM) to replace the fully connected layer, enabling the model to support inputs of any size and retain abundant information after multiple convolutions and pooling. Recently, the global average of the gradient is used in Grad-CAM to calculate the weights from the feature maps to overcome the limitations of CAM [30,31]. In Grad-CAM, the weight of the  $k$ th feature map of category  $c$  is defined as follows:

$$w_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k}, \quad (5)$$

where  $Z$  denotes the number of pixels in the feature map,  $y^c$  the score of the corresponding category  $c$ , and  $A_{ij}^k$  the pixel value at position  $(i, j)$  in the  $k$ th feature map. After obtaining the category weights of all feature maps, the weighted sum is calculated to obtain the attention map (or heat map). The attention map can be represented as

$$L^c = ReLU\left(\sum_k w_k^c A^k\right) \quad (6)$$

The attention map obtained using Grad-CAM increases the transparency of the CNN model. Although Grad-CAM provides the visualization of CNN for classification, it cannot accurately locate objects when the input images contain multiple objects in the same class [32,33]. Therefore, to identify object locations more accurately, the weight formulation connected to the feature map should be modified. The weights  $w_k^c$  independent of position  $(i, j)$  in Grad-CAM can be represented as shown in Eq. (5). In Grad-CAM, every sample is composed of multiple input image categories; therefore, its formulation can be used to identify the target location without the position  $(i, j)$ . However, the pixel position is critical for an input image containing multiple objects in the same class. In Grad-CAM++, the pixel locations are considered in the weights  $w_k^c$  [32,33], and the calculation formula is

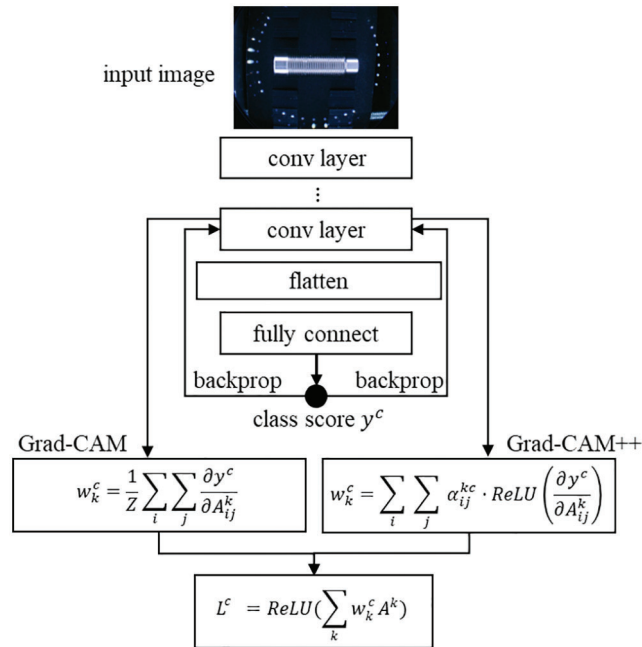
$$w_k^c = \sum_i \sum_j \alpha_{ij}^{kc} \cdot ReLU\left(\frac{\partial y^c}{\partial A_{ij}^k}\right), \quad (7)$$

where  $y^c$  denotes the score of class  $c$ , and  $\alpha_{ij}^{kc}$  denotes the gradient weights.  $\alpha_{ij}^{kc}$  can be obtained by partially differentiating  $y^c$  and  $A_{ij}^k$  twice, and  $\alpha_{ij}^{kc}$  can be represented as

$$\alpha_{ij}^{kc} = \frac{\frac{\partial^2 y^c}{(\partial A_{ij}^k)^2}}{2 \frac{\partial^2 y^c}{(\partial A_{ij}^k)^2} + \sum_a \sum_b A_{ab}^k \left\{ \frac{\partial^3 y^c}{(\partial A_{ij}^k)^3} \right\}}, \quad (8)$$

where  $(i, j)$  and  $(a, b)$  denote the same iterators in the  $k$ th feature map, and  $Z^{-1}$  is the solution for  $\alpha_{ij}^{kc}$  in Grad-CAM++. When  $\alpha_{ij}^{kc} = Z^{-1}$ , Grad-CAM++ reduces to Grad-CAM, i.e., Grad-CAM++ provides a general

formulation for obtaining the attention map of the input image. Grad-CAM and Grad-CAM++ exhibit the same structure despite their various differences, as shown in Fig. 6.



**Figure 6:** Differences between Grad-CAM and Grad-CAM++

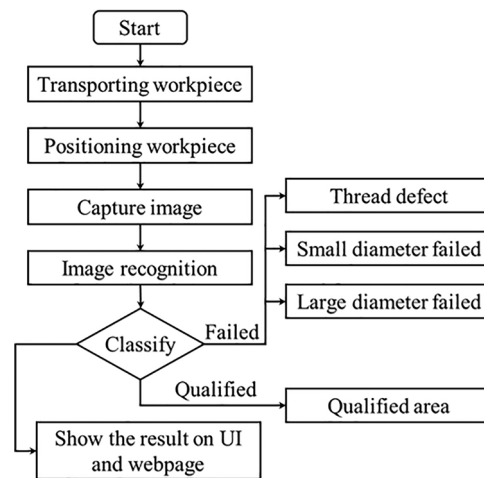
### 3.3 Defect Detection System

Fig. 7 shows the system flowchart of the proposed AOI defect detection. The screw is sent from the machine tool to the positioning mechanism by the feeding conveyor (shown in Fig. 3) and then placed on the detection conveyor. When the screw enters the classification area, the corresponding images from three perspectives are captured, and then measurement and defect detection are performed. The computer completes the image recognition and then sends the results to the classification area, where the screw is sorted. After the screw is sorted, the results are shown in the user interface (Fig. 8) and the webpage for monitoring (Fig. 9). The detection results and thread defects are updated at the interface after each inspection, and the images are updated simultaneously. Dimension measurements can assist in evaluating the tool-wear condition of the machine tool. The functions of real-time monitoring, alarm, and suspension systems are provided. The user interface shown in Fig. 8 presents the dimension measurements and defect detection results. In the defect detection results, the output value of a qualified thread is 1, and the output value of a failed thread is 0 (as shown in Fig. 9). The red lines indicate the range of acceptable tolerance. Both the interface and webpage can assist in determining whether the machine is operating abnormally.

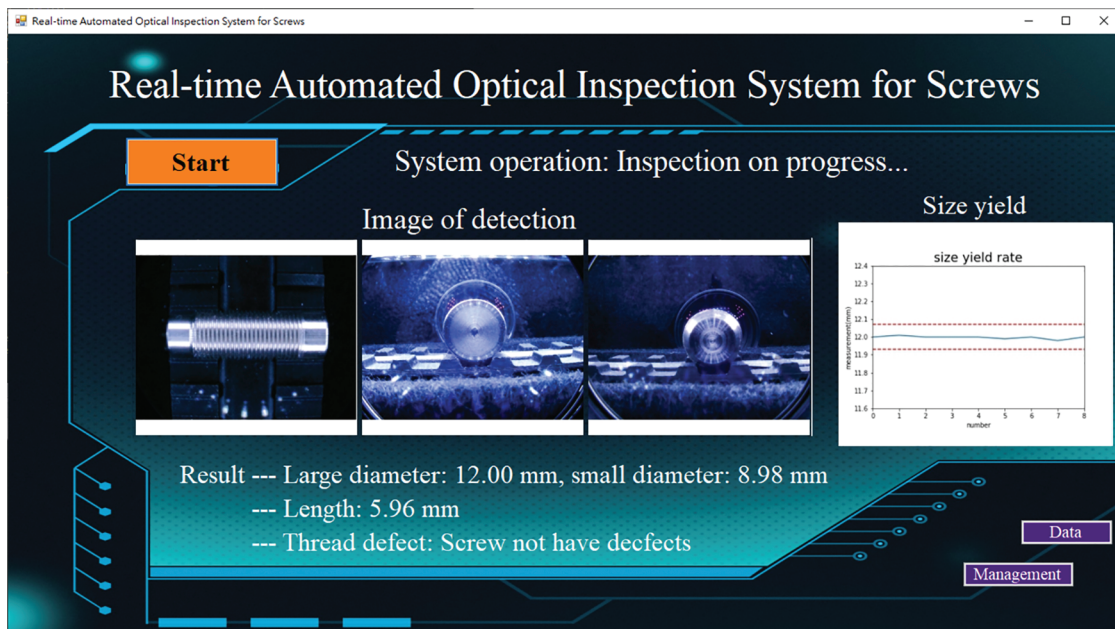
## 4 Experimental Results of Defect Detection

Herein, we consider a scenario in which the acquisition and labeling of numerous images is challenging. VGG16, which exhibits high feature extraction capabilities, was utilized and fine-tuned via transfer learning [29], and the corresponding structure is shown in Fig. 10. Because the number of defect samples is typically small, VGG16 with parameter transfer learning for defect detection was employed in this study. VGG16 comprises multiple convolutional layers, which endows it with better feature-extraction

capabilities compared with ordinary neural networks. VGG16 uses millions of images to classify thousands of categories through 13 convolutional layers and emphasizes the use of numerous  $3 \times 3$  filters in its convolutional layers. When larger filters are used instead of smaller filters, the receptive field is improved, which increases the amount of information obtained. In this study, the output layer of VGG16 was rewritten to classify the two categories regardless of whether a thread contained a defect.

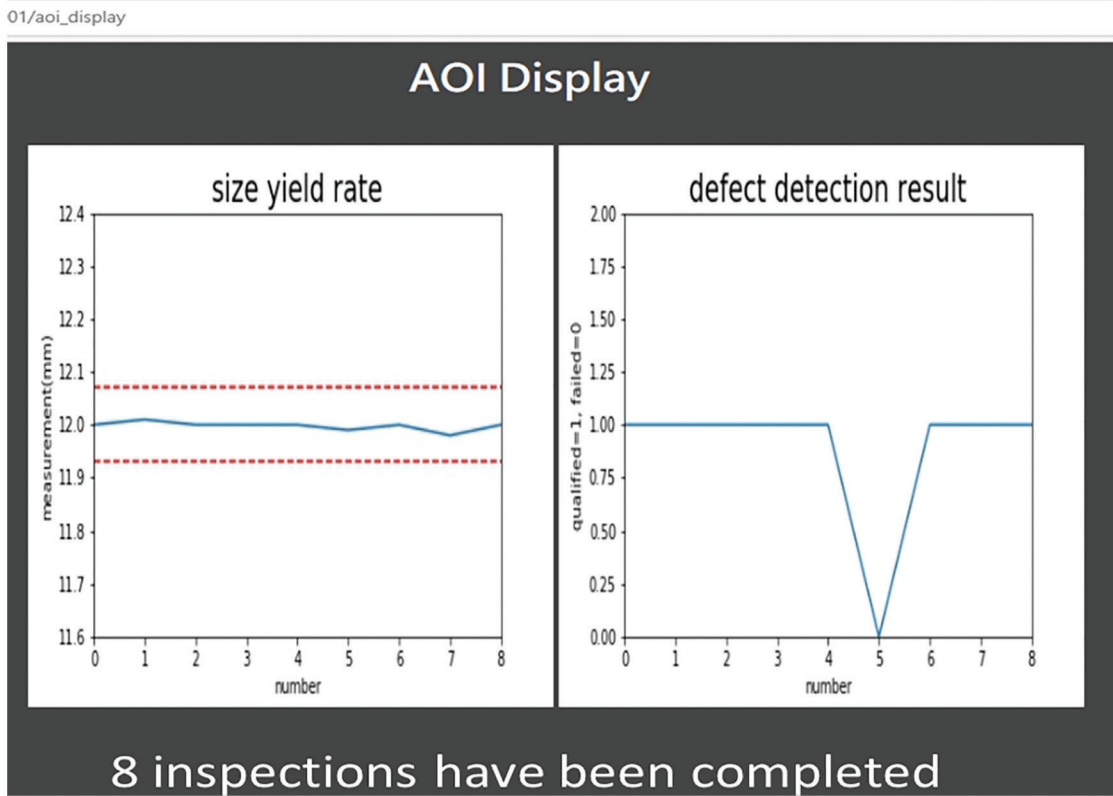


**Figure 7:** System flowchart for inspection detection



**Figure 8:** User interface of system



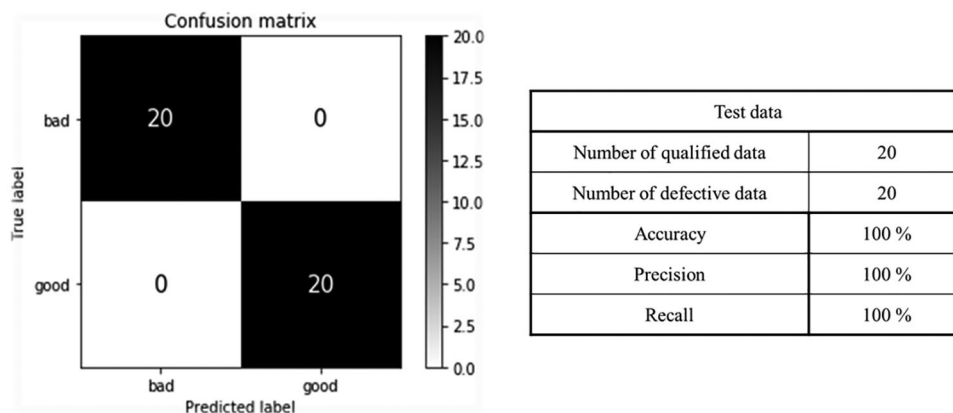


**Figure 9:** Webpage for monitoring machine tools

Hyper Parameters	
Epochs	1000
Batch size	512
Optimizer	adam
Loss	Binary crossentropy
Model Structure	
Input (224×224×3)	Fixed weights of VGG16 ImageNet (Non-trainable)
Conv3-64	
Conv3-64	
maxpool	
Conv3-128	
Conv3-128	
maxpool	
Conv3-256	
Conv3-256	
Conv3-256	
maxpool	
Conv3-512	
Conv3-512	
Conv3-512	
maxpool	
Flatten	
FC-512 (sigmoid)	Customized classifier (Trainable)
Dropout-0.4	
FC-512 (sigmoid)	
Dropout-0.4	
Output-2 (sigmoid)	

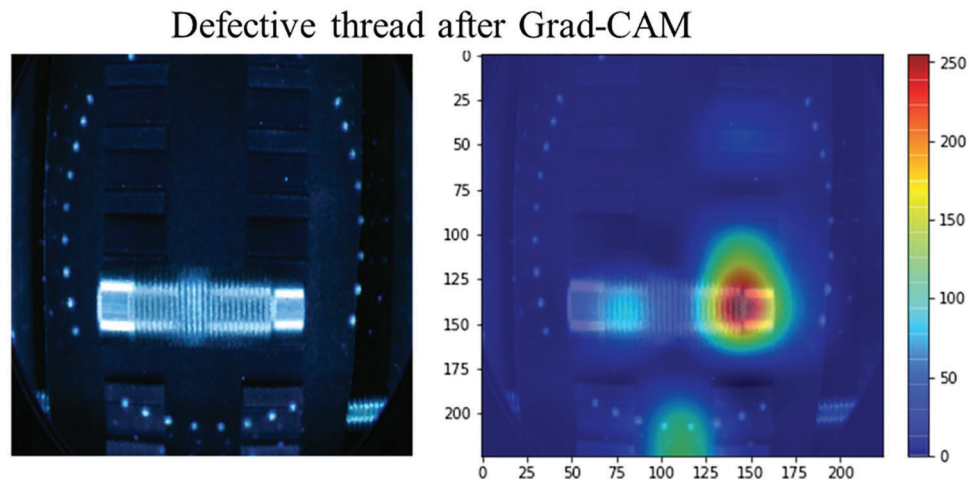
**Figure 10:** VGG16 structure [29]

In this study, an insufficient image was augmented via the translation, rotation, and flipping of the screw. The model was trained for 1000 generations, and the training data included 242 and 228 qualified and defective components. Furthermore, the model achieved 100% accuracy. The laptop computer used in the proposed system comprised an Intel Core i5-5200U processor, a GeForce 930M graphics processing unit, and 8 GB of random-access memory. Test data were obtained from the actual machining results, including those of 20 qualified and 20 defective components. To avoid data imbalance and insufficient data in the training model, only 40 screws were used for the test data. The confusion matrix presented in Fig. 11 indicates that 100% accuracy was achieved after the actual testing of the system. The corresponding accuracy, precision, and recall rates were all 100%. These results demonstrate the effectiveness of the proposed approach.

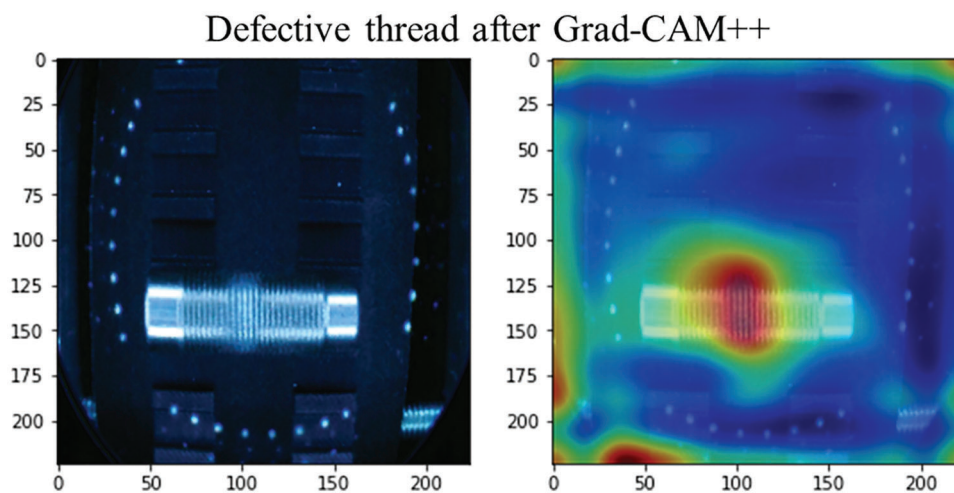


**Figure 11:** Confusion matrix of inspection results based on test data

As indicated above, the established model achieved high accuracy despite the low amount of training data used. To ensure that the model accurately identifies defects and facilitates in reverification, Grad-CAM methods were utilized to visualize the defect. The corresponding weight attention maps for defect detection calculated using Grad-CAM and Grad-CAM++ are presented in Figs. 12 and 13, respectively. As shown by the heat map in Fig. 12, the attention map prevents the defect region even when the classification accuracy is 100%, i.e., the attention map does not directly indicate defects by selecting the defect area. All test data exhibited the same phenomena. Therefore, Grad-CAM++ was adopted to improve the localization of the defects. The corresponding attention map obtained using Grad-CAM++ is shown in Fig. 13, where the defect region is denoted. Thus, the judgment of the model is reasonable. Based on this test, the effects of the weights  $w_k^c$  considering the pixel location can be observed. Grad-CAM++ not only confirms the rationality of the model, but also provides indicators for the entire thread.



**Figure 12:** Defective thread after performing Grad-CAM



**Figure 13:** Defective thread after performing Grad-CAM++

## 5 Conclusions

The proposed system was operated at minimal cost and integrated into a production line without the adjustment of existing processes. Using a five-megapixel camera, a dimensional tolerance of 0.012 mm was obtained, and a defect detection accuracy of 100% was achieved using a limited number of samples. Furthermore, the user interface and webpage can be used to monitor the production lines and equipment status. Additionally, the developed model was visualized using Grad-CAM and Grad-CAM++, and the results of the two methods were compared. Grad-CAM++ performed more effectively, provided a reasonable explanation for each classification result, and accurately located the thread defects. The proposed system resulted in a more complete process and increased consumer confidence as it was able to perform quality control within 20 s.

**Acknowledgement:** This study was supported partially by the Ministry of Science and Technology, Taiwan, under contracts MOST-110-2634-F-009-024, 109-2218-E-150-002, and 109-2218-E-005-015.

**Funding Statement:** The authors received no specific funding for this study.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] F. Hsu and C. Shen, "The design and implementation of an embedded real-time automated IC marking inspection system," *IEEE Transactions on Semiconductor Manufacturing*, vol. 32, no. 1, pp. 112–120, 2019.
- [2] D. Jeon, U. Jung, K. Park, P. Kim, S. Han *et al.*, "Vision-inspection-synchronized dual optical coherence tomography for high-resolution real-time multidimensional defect tracking in optical thin film industry," *IEEE Access*, vol. 8, pp. 190700–190709, 2020.
- [3] W. Chen, Y. Gao, L. Gao and X. Li, "A new ensemble approach based on deep convolutional neural networks for steel surface defect classification," in *Proc. 51st CIRP Conf. on Manufacturing Systems*, Stockholm, Sweden, vol. 72, pp. 1069–1072, 2018.
- [4] S. Mei, Q. Cai, Z. Gao, H. Hu and G. Wen, "Deep learning based automated inspection of weak microscratches in optical fiber connector end-face," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–10, 2021.
- [5] Q. Luo, X. Fang, L. Liu, C. Yang and Y. Sun, "Automated visual defect detection for flat steel surface: A survey," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 3, pp. 626–644, 2020.
- [6] K. Imoto, T. Nakai, T. Ike, K. Haruki and Y. Sato, "A CNN-based transfer learning method for defect classification in semiconductor manufacturing," *IEEE Transactions on Semiconductor Manufacturing*, vol. 32, no. 4, pp. 455–459, 2019.
- [7] H. Pan, Z. Pang, Y. Wang, Y. Wang and L. Chen, "A new image recognition and classification method combining transfer learning algorithm and mobilenet model for welding defects," *IEEE Access*, vol. 8, pp. 119951–119960, 2020.
- [8] S. Yan, B. Ma and C. Zheng, "A unified system residual life prediction method based on selected tribodiagnostic data," *IEEE Access*, vol. 7, pp. 44087–44096, 2019.
- [9] L. Hao, L. Bian, N. Gebraeel and J. Shi, "Residual life prediction of multistage manufacturing processes with interaction between tool wear and product quality degradation," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 1211–1224, 2017.
- [10] P. Ding, Q. Qian, H. Wang and J. Yao, "A symbolic regression based residual useful life model for slewing bearings," *IEEE Access*, vol. 7, pp. 72076–72089, 2019.
- [11] Z. Liu, H. Wang, W. Chen, J. Yu and J. Chen, "An incidental delivery based method for resolving multirobot pairwised transportation problems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 7, pp. 1852–1866, 2016.
- [12] B. Hichri, J. C. Fauroux, L. Adouane, I. Doroftei and Y. Mezouar, "Design of cooperative mobile robots for co-manipulation and transportation tasks," *Robotics and Computer-Integrated Manufacturing*, vol. 57, no. 3, pp. 412–421, 2019.
- [13] L. Hawley and W. Suleiman, "Control framework for cooperative object transportation by two humanoid robots," *Robotics and Autonomous Systems*, vol. 115, no. 5–6, pp. 1–16, 2019.
- [14] I. G. Plaksina, G. I. Chistokhina and D. V. Topolskiy, "Development of a transport robot for automated warehouses," in *2018 Int. Multi-Conf. on Industrial Engineering and Modern Technologies (FarEastCon)*, Vladivostok, Russia, pp. 1–4, 2018.
- [15] T. Nishi, S. Akiyama, T. Higashi and K. Kumagai, "Cell-based local search heuristics for guide path design of automated guided vehicle systems with dynamic multicommodity flow," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 2, pp. 966–980, 2020.
- [16] Z. Rozsa and T. Sziranyi, "Obstacle prediction for automated guided vehicles based on point clouds measured by a tilted LIDAR sensor," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 8, pp. 2708–2720, 2018.
- [17] Y. Liu, Z. Hou, Y. Tan, H. Liu and C. Song, "Research on multi-AGVs path planning and coordination mechanism," *IEEE Access*, vol. 8, pp. 213345–213356, 2020.

- [18] B. Lyons, G. Bierie and A. Marti, "Belt conveyor training: Changing behaviors, reducing risk, improving the bottom line," in *2015 IEEE-IAS/PCA Cement Industry Conf. (IAS/PCA CIC)*, Toronto, ON, Canada, pp. 1–5, 2015.
- [19] D. Qu, T. Qiao, Y. Pang, Y. Yang and H. Zhang, "Research on ADCN method for damage detection of mining conveyor belt," *IEEE Sensors Journal*, vol. 21, no. 6, pp. 8662–8669, 2021.
- [20] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed., Pearson Education international, Upper Saddle River, NJ, 2008.
- [21] Y. Liu, J. Guo and J. Yu, "Contrast enhancement using stratified parametric-oriented histogram equalization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 6, pp. 1171–1181, 2017.
- [22] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.
- [23] K. Ashwini and S. B. Rudraswamy, "Automated inspection system for automobile bearing seals," *Materials Today: Proceedings*, vol. 46, no. 10, pp. 4709–4715, 2021.
- [24] S. Fotouhi, F. Pashmforoush, M. Bodaghi and M. Fotouhi, "Autonomous damage recognition in visual inspection of laminated composite structures using deep learning," *Composite Structures*, vol. 268, no. 3, pp. 113960, 2021.
- [25] W. W. Fan and C. H. Lee, "Classification of imbalanced data using deep learning with adding noise," *Journal of Sensors*, vol. 2021, no. 1, pp. 1–18, 2021.
- [26] Y. He, K. Song, Q. Meng and Y. Yan, "An end-to-end steel surface defect detection approach via fusing multiple hierarchical features," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 4, pp. 1493–1504, 2020.
- [27] B. Shi and Z. Chen, "A layer-wise multi-defect detection system for powder bed monitoring: lighting strategy for imaging, adaptive segmentation and classification," *Materials & Design*, vol. 210, no. 1–4, pp. 110035, 2021.
- [28] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *2015 Int. Conf. on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.
- [30] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh *et al.*, "Grad-CAM: visual explanations from deep networks via gradient-based localization," in *2017 IEEE Int. Conf. on Computer Vision (ICCV)*, Venice, Italy, pp. 618–626, 2017.
- [31] H. Y. Chen and C. H. Lee, "Vibration signals analysis by explainable artificial intelligence (XAI) approach: Application on bearing faults diagnosis," *IEEE Access*, vol. 8, pp. 134246–134256, 2020.
- [32] A. Chattopadhyay, A. Sarkar, P. Howlader and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *2018 IEEE Winter Conf. on Applications of Computer Vision (WACV)*, Lake Tahoe, NV, USA, pp. 839–847, 2018.
- [33] Y. R. Lin, C. H. Lee and M. C. Lu, "Robust tool wear monitoring system development by sensors and feature fusion," *Asian Journal of Control*, vol. 6, no. 3, pp. 1–17, 2022.
- [34] M. D. Zeiler, G. W. Taylor and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *2011 Int. Conf. on Computer Vision*, Barcelona, Spain, pp. 2018–2025, 2011.
- [35] K. H. Sun, H. Huh, B. A. Tama, S. Y. Lee, J. H. Jung *et al.*, "Vision-based fault diagnostics using explainable deep learning with class activation maps," *IEEE Access*, vol. 8, pp. 129169–129179, 2020.
- [36] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva and A. Torralba, "Learning deep features for discriminative localization," in *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 2921–2929, 2016.