


Article

Tire Bubble Defect Detection Using Incremental Learning

Chuan-Yu Chang , You-Da Su and Wei-Yi Li

Department of Computer Science and Information Engineering, National Yunlin University of Science and Technology, Douliou 64002, Taiwan

* Correspondence: chuanyu@yuntech.edu.tw

Abstract: Digital shearography is a technique that has recently been applied to material inspections that cannot be performed by the naked eyes, including the detection of air bubble defects in tires. Although digital shearography detects bubbles that are not visible to the naked eyes, the process of determining tire defects still relies on field operators, with inconsistent results depending on the experiences of the field operator personnel. New or different types of bubble defects that AI models have not previously recognized are often missed, resulting in an inadequate quality detection model. In this paper, we propose a bubble defect detection method based on an incremental YOLO architecture. The data for this research was provided by the largest tire manufacturer in Taiwan. In our research, we classify the defects into six distinct categories, pre-process the images to allow better detections of less-noticeable defects, increase the amount of training data used, and generate an initial training model with the YOLO framework. We also propose an incremental YOLO method using small-model training for previously unobserved defects to improve the model detection rate. We have observed detection accuracy and sensitivity of 98% and 90% in the experimental results, respectively. The methods proposed in this paper can assist tire manufacturers in achieving semi-automatic quality inspections and labor cost reductions.



Citation: Chang, C.-Y.; Su, Y.-D.; Li, W.-Y. Tire Bubble Defect Detection Using Incremental Learning. *Appl. Sci.* **2022**, *12*, 12186.

<https://doi.org/10.3390/app122312186>

Academic Editors: Tibor Krenicky, Maros Korenko and Juraj Ruzbarsky

Received: 10 November 2022

Accepted: 24 November 2022

Published: 28 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: tire bubble defect detection; digital shearography; YOLO; deep learning

1. Introduction

Industry 4.0 enables data communication, collection and analysis by machines, enabling faster, more agile and efficient processes to manufacture quality products with minimal expenditure. Since 2012, researches on computer vision tasks based on convolutional neural networks have progressed fairly rapidly [1,2].

During the tire manufacturing process, air may be trapped in the tire to form bubbles. The bubbles can compromise the tire's structural integrity, leading to the risk of a punctured tire if the pressure is not distributed evenly. Therefore, tire manufacturers inspect the tires for bubble defects before they are sold to reduce the risk of punctures. The tire bubbles are difficult to detect by the naked eye since they are internal defects. The operator puts the tire into the machine, and air bubbles in the tire cause the tire surface to bulge in the vacuum. Digital shearography [3] is used to obtain speckle images of the tire at different angles for manual inspections. However, tired or inexperienced inspectors can lead to inaccuracies and inconsistency in the inspection outcomes.

The tire defect detection process is extremely labor and time-intensive, even though it is only a small part of the quality control process in the tire manufacturing industry. Many difficulties arise during the inspection due to the inconsistent and subjective assessments by inspectors with different experiences. To ensure product quality and reduce labor costs, deep learning has been gradually incorporated in the industry by many manufacturers to reduce labor-intensive work and improve production efficiency effectively. With the help of artificial intelligence, operators only need to confirm the bubble defects detected by machine, which can reduce the workload of labors. Manufacturers do not need to recruit a

lot of labor to detect bubble defects one by one; developments in automation technology can also address the problems caused by inconsistent inspection standards.

In our previous research [4–6], we divided the entire speckle image into several blocks, and a convolutional neural network is used to detect the tread bubbles and the sidewall separately. However, there is still room for improvement regarding the detection speed and detection rate. In this paper, we use an object detection method to improve the bubble defect detection speed and detection rate. In addition, to address the situation where the model developed in the laboratory is actually introduced into the field with a reduced defect detection rate, we propose an incremental YOLO (You Only Look Once) architecture to add on-site misidentified samples to model training to improve the model’s flaw detection accuracy on the actual site. The YOLO machine learning algorithm uses features learned by a deep convolutional neural network to detect objects in real time. YOLO has the advantage of being much faster than other networks and still maintaining high accuracy. Experimental results show that the improved YOLOv3 SPP outperforms other YOLO series architectures (Tiny YOLO, YOLOv3 and YOLOv3 SPP) in bubble defect detection [7–10]. Furthermore, the proposed incremental learning also demonstrates the effectiveness of improving the detection accuracy of bubble defects in real field.

The rest of this paper is organized as follows. First, the proposed method is described in detail in Section 2. Then, we present the experimental results and discuss the results in Section 3 and, finally, the conclusions are delivered in Section 4.

2. Proposed Method

Since the tire factory cannot provide enough defect images for model training, in practice, it often happens that the recognition rate of the model trained in the laboratory is greatly reduced when it is actually imported into the field. In order to shorten the time for importing the model into the field, this paper proposes an incremental learning method for bubble detection on tire tread and sidewalls. We test the lab-trained model with images from the tire production site and collect misidentified samples, incrementally train by freezing the Yolo Backbone, and only re-train the Neck and Prediction network layers to improve defect detection rates [11]. The main steps are described as follows:

1. Manually mark the position and type of the bubble and classify it into one of the six types;
2. Perform image enhancement and increase the number of training samples;
3. After resizing, use the YOLO architecture proposed in this paper for model training;
4. Overlay the detected defects on the original image for visualization.

We apply incremental learning to the inspection process to account for potentially overlooked defects. Incremental learning allows small models to be trained on the missed defects and improves the model’s accuracy. The process is as shown in Figure 1.

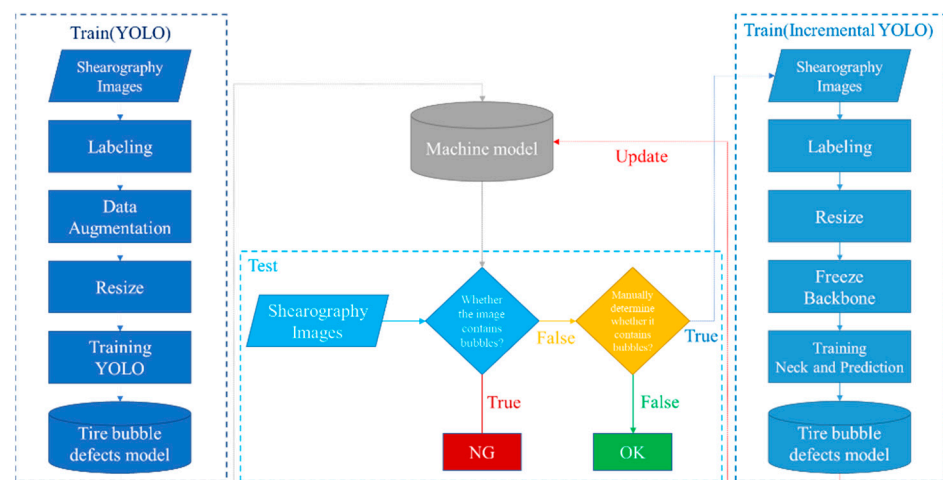


Figure 1. The bubble detection process.

2.1. Data Sample Categorization and Enhancement

Speckle images with dimensions of 1360×1024 pixels are used as input for inspection. Defects can be classified into different categories based on their location, size and shape. Using only a binary classification to separate images with and without bubbles would likely result in higher misjudgment rates. In this research, we divided the bubble defects into six categories, as listed below in Figure 2.

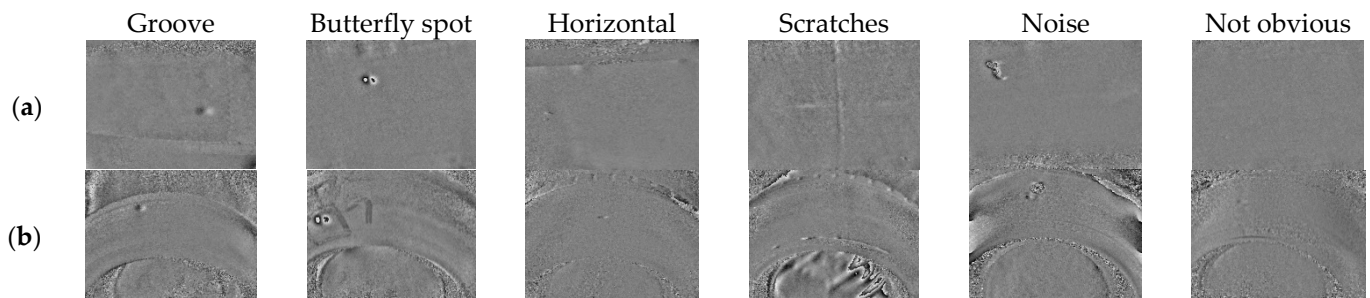


Figure 2. Six types of bubble images (a) Tire tread defects and (b) Tire sidewall defects.

To improve the accuracy for the hard-to-spot defects that are likely to be overlooked during the inspection, we used an image enhancement technique called Contrast Limited Adaptive Histogram Equalization (CLAHE) [12] to increase the visibility of the subtle bubbles present in the image. As shown in Figure 3, the bubbles in the red frame are much more obvious than the original image.

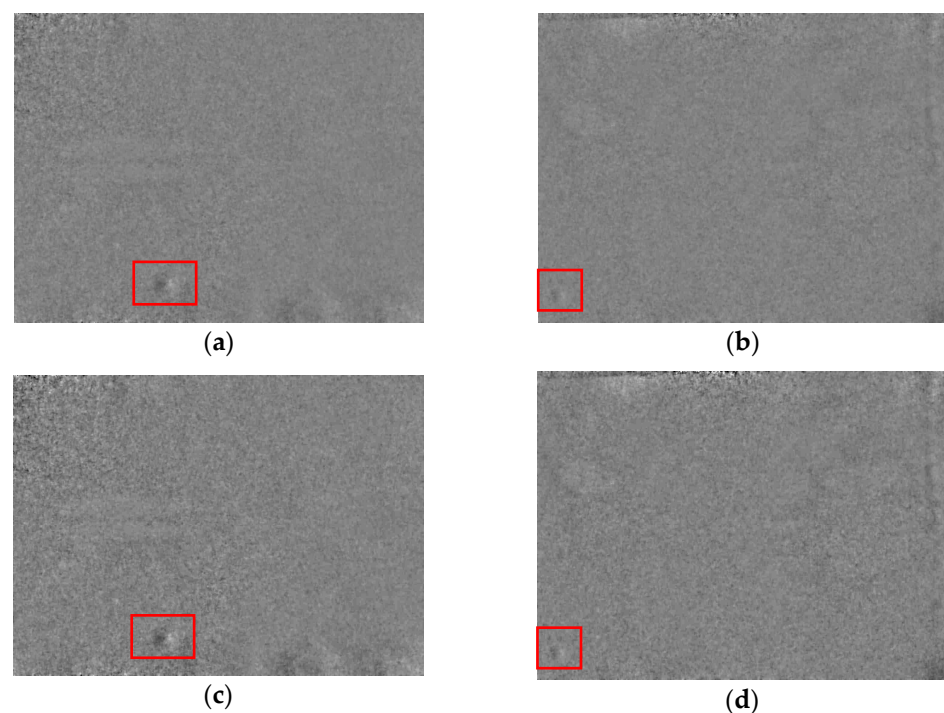


Figure 3. Image enhancement (a,b) original speckle images, (c,d) Contrast enhanced images.

As deep learning network training requires a large amount of sample data to attain a higher detection rate, insufficient defect samples in the dataset provided by the tire manufacturer may affect the effectiveness of the model. To address this issue, we combined the original and enhanced images in the model training set to increase sample data [13].

2.2. Image Resize

Since the sizes of the grids in the YOLOv3 architecture [8] are 32×32 , 16×16 and 8×8 , we resize the 1360×1024 input images to 1344×1024 so that the dimensions are multiples of 32. For computational convenience, we further divided the entire speckle image into multiple 480×480 sub-images. Figure 4 shows what the bubbles look like at three different scales (32×32 , 16×16 and 8×8). Obviously, the area of the bubbles is smaller than the 32×32 grid. The finer grids allow for more features in the bubble defects to be learned.

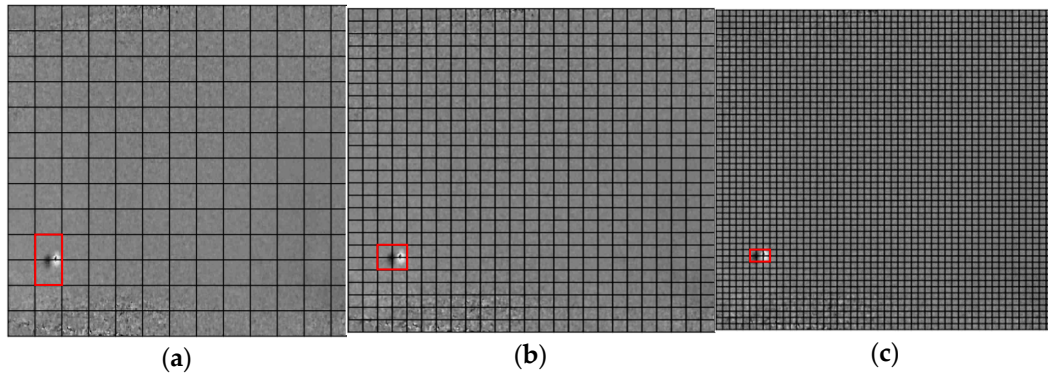


Figure 4. Visualization of different grid sizes: (a) Large scale (32×32), (b) Medium scale (16×16), and (c) Small scale (8×8).

2.3. Framework: Modify the YOLOv3 SPP Architecture

The YOLOv3 architecture [8] is mainly divided into four sections: the Input terminal, Backbone, Neck and Prediction, as shown in Figure 5. The functions of each section are given below,

1. Input terminal: Image pre-processing and data augmentation;
2. Backbone: Feature extraction for lower-level, shallow features which are relatively similar, including edges, colors and textures;
3. Neck: Feature enhancement conducted by processing and enhancing the shallow features extracted by the backbone and allowing the model to learn the characteristics of tire bubbles.
4. Prediction: Outputting a final prediction containing the loss function and the Non-Max Suppression Algorithm of the bounding and anchor boxes.

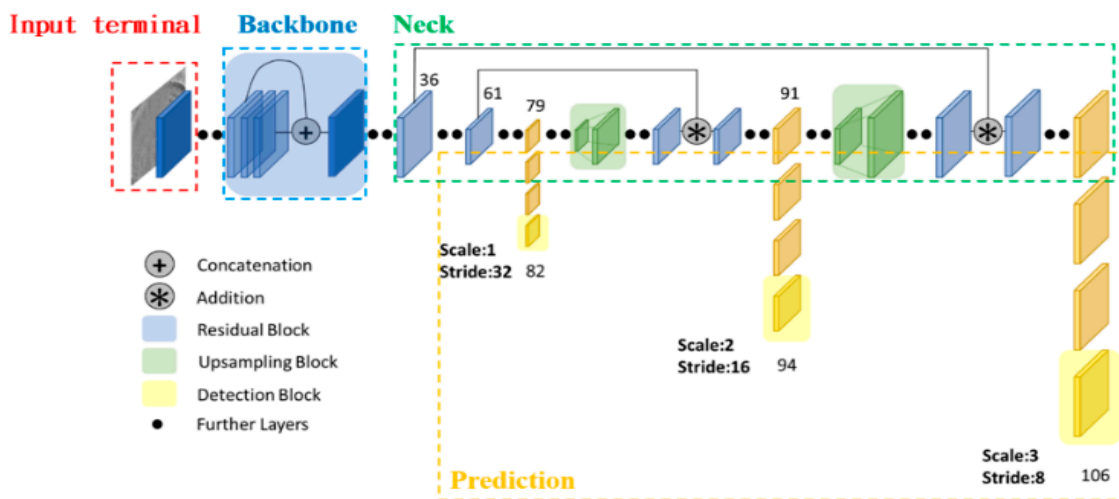


Figure 5. Four main sections of the YOLOv3 architecture [8].

The original YOLOv3 SPP [14] uses Darknet 53 as the backbone with a three-scale detection network. Since the bubble defects only occupy a small portion of the speckle image, therefore, we modify the YOLOv3 SPP architecture, adjust it to a two-scale output and modify the backbone network. As shown in Figure 6, we removed the large-scale network layers in the Darknet 53 architecture, adding the same number of layers back to the small-scale network layer to improve small-scale feature acquisition. As the large-scale sections consist of many channels, removing the layers from the backbone network reduces the number of parameters by nearly 1/3, significantly reducing the required computing resources. The FPN [11] structure is referenced for re-joining the backbone network and the prediction layer. Furthermore, we have found that the addition of the SPP method effectively increases the range of backbone features and significantly separates the most important context features. The addition of a layer of SPP structure after up-sampling achieves two-scale local and global feature fusion.

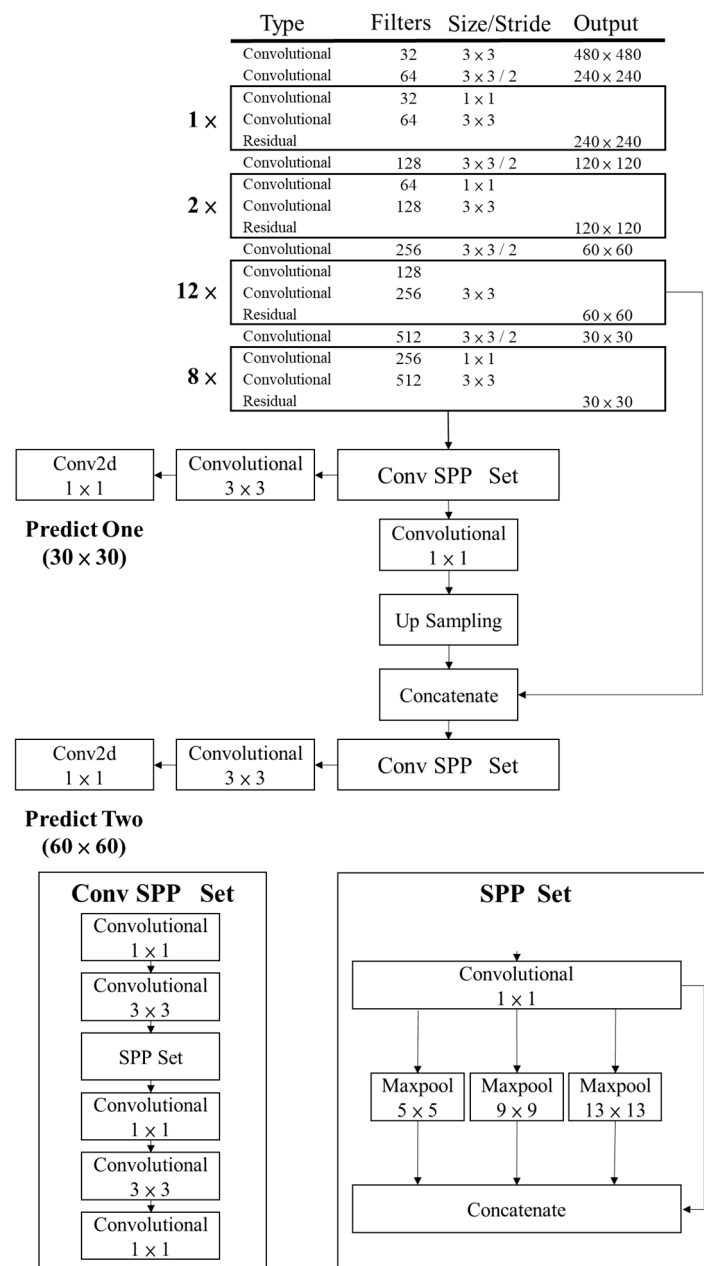


Figure 6. The YOLOv3 SPP two-scale output network architecture.

2.4. The Incremental YOLO Architecture

Previous methods merge new data into the training datasets before retrain to reduce missed detections due to incorrect identification of defective images by the original model. Although this method improves the detection rate of the model, it is a time-consuming process. We propose an incremental YOLO approach with the concept of transfer learning. Since we can only train the object detection model on images with objects, we have manually relabeled and added the misidentified images to the incremental YOLO training dataset. For incremental training, we retained the shallow features in the original model while training the deeper features of the network layer. As a result, the bubble defects missed during inspection accumulate and are used to rapidly update the model to improve the detection rate.

We froze the Backbone section for incremental training and only trained the Neck and Prediction network layers to improve the detection rate. Since the backbone network mainly extracts low-level features such as edges, colors, and textures of images. The neck layer is mainly to process and enhance the low-level features extracted by the backbone, so that the model can learn features suitable for tire bubbles. To reduce the computational load of feature extraction and efficiently learn the image features of the field, we froze the backbone layer and retrained the neck layer in incremental learning to strengthen the bubble features of the field. As shown in Figure 7, the blue box indicates the frozen Backbone section, while the red box indicates the sections where the incremental YOLO architecture performs parameter adjustments.

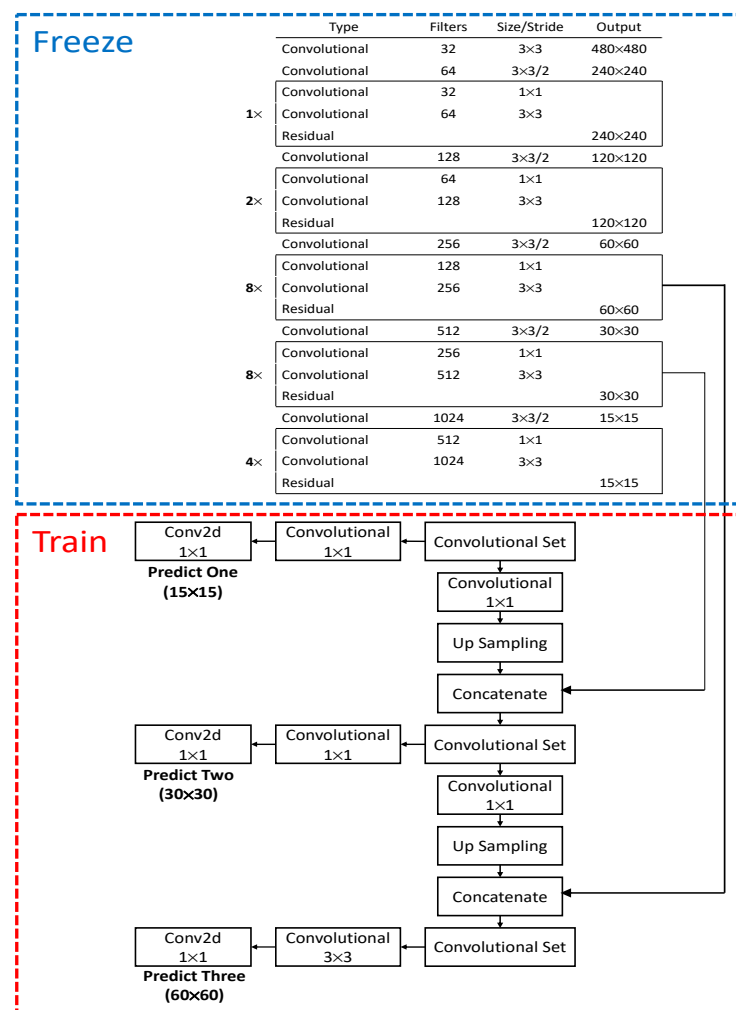


Figure 7. Steps of the incremental learning method.

3. Experimental Results

3.1. Dataset and Experimental Environment

The tire speckle images used in this paper were provided by Taiwan's largest tire manufacturer. The dataset consists of tire tread and sidewall images taken from different angles. Each set contains 24 to 26 images of size 1360×1024 pixels. One tire speckle image may include more than one bubble. The training and the testing dataset are independent. The training dataset contains 2450 images with 3116 bubble defects. The dataset is classified into six defect categories, as listed in Table 1. The sample tire tread images consist mostly of grooved bubbles, with 1421 images in the category, followed by butterfly-shaped bubbles. The remaining categories only account for a small part of the overall training data. The dataset consists of 1763 tread samples, containing a total of 2247 bubbles. A similar pattern exists for the sidewall images, with 321 grooved images containing 377 grooved bubbles, followed by butterfly-shaped bubbles and images from the remaining categories. There are altogether 687 sidewall samples containing 869 bubbles. Altogether, the dataset contains 2450 training images and a total of 3116 bubbles.

Table 1. Training Dataset.

Data Class	Samples		Bubbles	
	Tread	Sidewall	Tread	Sidewall
Groove	1156	321	1421	377
Butterfly spot	322	265	431	347
Horizontal	91	22	134	49
Scratches	88	10	137	17
Noise	45	61	48	69
Not obvious	61	8	76	10
Total	1763	687	2247	869
Training	2450		3116	

The testing dataset is described in Table 2. Binary classification of normal and defective images is used in this study, as the primary purpose is to detect defective bubbles without ensuring the correct categorization. The test data contains 500 normal tread sample images and 810 tread images with bubble defects, as well as 500 normal sidewall sample images and 505 sidewall images with bubble defects.

Table 2. Testing Dataset.

	Data Class	# of Image
	Tread	Non-bubbles
Bubbles		810
Sidewall	Non-bubbles	500
	Bubbles	505

The experiments were performed in a Windows environment, using an Intel Core i7-8700K processor and a GeForce GTX 1080 Ti GPU with the Python Keras framework.

We used four evaluation methods to verify the results of this experiment: Accuracy, Sensitivity, Specificity and Precision. The equations are as follows;

$$\text{Accuracy} = (TP + TN)/(P + N) \quad (1)$$

$$\text{Sensitivity} = TP/P \quad (2)$$

$$\text{Specificity} = TN/N \quad (3)$$

$$\text{Precision} = TP/(TP + FP) \quad (4)$$

In the above equations, P represents the total number of images containing bubble defects, N represents the total number of images without defects. TP represents the number of true positives, which are the correctly identified images with defects, and TN represents the number of true negatives, which are the correctly identified images without defects. FP represents the number of false positives, which are images without defects and incorrectly marked as defective, and FN represents the number of false negatives, which are images with defects and incorrectly marked as normal. As the purpose of quality control in tire manufacturing is to detect all bubble defects, we use sensitivity as the main evaluation standard for this paper since false positives have a lesser effect compared to false negatives.

In order to verify that the proposed modified YOLOv3 SPP architecture is superior to other YOLO series architectures in bubble defect detection, this experiment will compare Tiny YOLO, YOLOv3, YOLOv3 SPP and other methods. In addition, we will also verify through experiments that the incremental learning method proposed in this paper can effectively improve the bubble defects detection accuracy on the actual site.

3.2. Results Obtained by the Tiny YOLO Architecture

The Tiny-YOLO architecture is a YOLO-based object detector [7]. It is designed to contain a simpler network structure and utilize fewer computing resources. From Table 3, the sensitivity of the Tiny YOLO architecture is 79.77%. The incremental Tiny YOLO architecture achieves a better bubble detection rate by an additional 5%.

Table 3. Results Obtained by (a) the Tiny YOLO Architecture, (b) the incremental Tiny YOLO Architecture.

Result	(a)		(b)	
	Tread	Sidewall	Tread	Sidewall
TP	638	411	690	424
TN	496	486	500	2
FP	4	14	0	8
FN	172	94	120	81
Sensitivity	79.77%		84.71%	

3.3. Results Obtained by the YOLOv3 Architecture

Experimental results in Table 4 show that the detection rates of the YOLOv3 architecture. For tire sidewall images, a larger input size has better recognition. We have noticed that the increased small-scale output of the YOLOv3 architecture improves the detection rate by an average of 2% compared to the Tiny YOLO architecture. This also proves that the model is more sensitive to bubble features at a smaller scale during the feature extraction stage. Adding the incremental YOLO architecture can further increase the detection rate by about 6%.

Table 4. Results Obtained by (a) the YOLOv3 Architecture, (b) the Incremental YOLOv3 Architecture.

Result	(a)		(b)	
	Tread	Sidewall	Tread	Sidewall
TP	650	419	722	435
TN	499	496	497	490
FP	1	4	3	10
FN	160	86	88	70
Sensitivity	81.29%		87.98%	

3.4. Results Obtained by the YOLOv3 SPP Architecture

Spatial Pyramid Pooling (SPP) [14] is a pooling layer, using 1×1 , 5×5 , 9×9 , and 13×13 maximum pooling methods to achieve multi-scale fusions verification including local and global features. In this experiment, we added an SPP layer to the YOLOv3 backbone network to improve the detection accuracy at the cost of higher computing resource requirements.

Table 5 shows the detection results after adding an SPP structure to the YOLOv3 architecture to achieve local and global feature fusions. The results show that both tread and sidewall samples have higher detection accuracies than those obtained by the YOLOv3 architecture. For both the sidewall and the tread, after adding the SPP structure, the detector extracts more feature information than the YOLOv3 architecture in the feature extraction stage, which effectively increases the reception range of the backbone feature and significantly improves the detection rate. Adding the incremental YOLO architecture further increases the detection rate by approximately 6%.

Table 5. Results Obtained by (a) the YOLOv3 SPP Architecture, (b) the Incremental YOLOv3 SPP Architecture.

Result	(a)		(b)	
	Tread	Sidewall	Tread	Sidewall
TP	714	431	770	459
TN	497	492	489	477
FP	3	8	11	23
FN	96	74	40	46
Sensitivity	87.07%		93.46%	

3.5. Results Obtained by the Modified YOLOv3 Architecture

Although the results obtained by the modified YOLOv3 architecture are slightly lower when compared to those obtained by the YOLOv3 SPP architecture, we have observed a slightly improved detection rate relative to the YOLOv3 architecture in both the tread and sidewall defect detections, as seen by comparing Tables 4 and 6. The modified scale prediction network layer improves the detection rates and decreases the number of outputs, since reducing the large-scale feature extraction and strengthening the small-scale feature extraction increase the extracted bubble features. For the tire bubble detection, similarly, the two-output modification is more suitable for verification. The incremental YOLO architecture further increases the detection rate by about 4%.

Table 6. Results Obtained by (a) the Modified YOLOv3 Architecture, (b) the Incremental YOLOv3 Architecture.

Result	(a)		(b)	
	Tread	Sidewall	Tread	Sidewall
TP	668	442	736	431
TN	500	490	499	485
FP	0	10	1	15
FN	142	63	74	74
Sensitivity	84.81%		88.75%	

3.6. Results Obtained by the Modified YOLOv3 SPP Architecture

Table 7 shows the experimental results of the proposed architecture. The proposed architecture has achieved higher accuracies than the YOLOv3 SPP architecture in the tire tread and sidewall detections due to the backbone network modifications, where large-scale feature extraction layers were removed, and small-scale layers were strengthened to allow the network to learn detailed defect features. Verifying results using two outputs instead

of three outputs also resulted in better accuracy. By adding the SPP structure before the two outputs, the detection rate of the architecture is increased significantly while reducing nearly 40 million parameters. Adding the incremental YOLO architecture further increases the detection rate by about 6%.

Table 7. Results Obtained by (a) the Modified YOLOv3 SPP Architecture, (b) the Incremental YOLOv3 SPP Architecture.

Result	(a)		(b)	
	Tread	Sidewall	Tread	Sidewall
TP	758	459	801	495
TN	490	487	481	432
FP	10	13	19	68
FN	52	46	9	10
Sensitivity	92.54%		98.56%	

3.7. Comparison Results of Different YOLO Architectures

Table 8 summarizes the results of the five architectures. Compared to the Tiny YOLO architecture, the YOLOv3 architecture has performed better in detection. Similarly, the modified YOLOv3 architecture performed slightly better than the original YOLOv3 detector. The SPP structure in the YOLOv3 SPP architecture has also significantly improved the detection rates of the YOLOv3 architecture. Finally, the modified YOLOv3 SPP architecture proposed in this paper has achieved the highest accuracy, with a 5.5% increase in accuracy compared to the unmodified YOLOv3 SPP model.

Table 8. Results Obtained by Different Architecture (a) Tiny YOLO (b) YOLOv3 (c) YOLOv3 SPP (d) Modified YOLOv3 (e) Proposed modified YOLOv3 SPP Model.

Result	Tread				Sidewall				Sensitivity
	TP	TN	FP	FN	TP	TN	FP	FN	
(a)	638	496	4	172	411	486	14	94	79.77%
(b)	650	499	1	160	419	496	4	86	81.29%
(c)	714	497	3	96	431	492	8	74	87.07%
(d)	668	500	0	142	442	490	10	63	84.81%
(e)	758	490	10	52	459	487	13	46	92.54%

3.8. Comparison of the Different Incremental YOLO Architectures

Table 9 lists the comparison results of the above five architectures in conjunction with the incremental learning. The proposed improved YOLOv3 SPP method achieves the highest sensitivity among the results obtained using the incremental learning. The proposed model's detection rate is 6% higher than the modified YOLOv3 SPP method without the incremental YOLO architecture (see Table 8), significantly reducing the number of false negatives in defect detection. The proposed method's detection time is shorter than all other methods except the Tiny YOLO architecture. Nevertheless, despite the Tiny YOLO architecture's simple, easy-to-use network structure and low computing resource requirements, it is unsuitable for the detection task due to its poor detection rates. Hence, our proposed method provides a semi-automated defect detection approach for tire manufacturers, effectively reducing labour costs and improving inspection accuracy.

Table 9. Results and detection times of the incremental YOLO architecture with different architectures (a) Tiny YOLO (b) YOLOv3 (c) YOLOv3 SPP (d) Modified YOLOv3 (e) Proposed Model.

Result	Tread				Sidewall				Sensitivity	Time
	TP	TN	FP	FN	TP	TN	FP	FN		
(a)	690	500	0	120	424	2	8	81	84.71%	70(s)
(b)	722	497	3	88	435	490	10	70	87.98%	132(s)
(c)	770	489	11	40	459	477	23	46	93.46%	134(s)
(d)	736	499	1	74	431	485	15	74	88.75%	133(s)
(e)	801	481	19	9	495	432	68	10	98.56%	125(s)

4. Conclusions

This paper proposes a semi-automated tire defect detection process using a dataset of speckle images provided by the largest tire manufacturer in Taiwan, and labeled by professionals into six defect categories. To supplement the dataset, we have enhanced images of defects that are difficult to detect and added them to the dataset during training to improve accuracy. Using the training sample, we obtained a detection rate of 92.54% with the initial model in the first stage. During the second stage, we implemented an incremental YOLO architecture upon the architecture in the first stage, which improved the detection rate to 98.56%, reduced the detection speed, and provided better detection rates than previous studies for the hard-to-detect bubbles. During the quality control process in tire manufacturing, the model is likely to encounter new defects that were not present in the training data. We can incrementally train the model to recognize new defects with the proposed incremental YOLO architecture, which significantly improves the accuracy compared to the original model. For defect recognition, the original and incremental models had misjudged rates of 2.3% and 7.7%, respectively. The misjudgments are primarily due to less visible defects in the speckle images. As a result, more research may be required to further improve the detection rate.

Author Contributions: Conceptualization, C.-Y.C. and Y.-D.S.; methodology, C.-Y.C.; software, W.-Y.L.; validation, C.-Y.C. and Y.-D.S.; investigation, W.-Y.L.; writing—original draft preparation, W.-Y.L. and Y.-D.S.; writing—review and editing, C.-Y.C. and Y.-D.S.; visualization, W.-Y.L.; supervision, C.-Y.C. and Y.-D.S.; project administration, C.-Y.C.; funding acquisition, C.-Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Intelligent Recognition Industry Service Center, a Featured Research Center of the Taiwan Ministry of Education's Higher Education SPROUT Project. And the APC was funded by Ministry of Education, Taiwan.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Bhat, S.S.; Selvam, V.; Ansari, G.A.; Ansari, M.D.; Rahman, M.H. Prevalence and Early Prediction of Diabetes Using Machine Learning in North Kashmir: A Case Study of District Bandipora. *Comput. Intell. Neurosci.* **2022**, *2020*, 2789760. [[CrossRef](#)] [[PubMed](#)]
- Gaddam, D.K.R.; Ansari, M.D.; Vuppala, S.; Gunjan, V.K.; Sati, M.M. Human Facial Emotion Detection Using Deep Learning. In *ICDSMLA 2020*; Springer: Singapore, 2021; pp. 1417–1427.
- Steinchen, W.; Yang, L. *Digital Shearography: Theory and Application of Digital Speckle Pattern Shearing Interferometry*; SPIE Press: Bellingham, DC, USA, 2003.
- Chang, C.-Y.; Huang, J.-K. Tires Defects Detection Using Convolutional Neural Networks. In Proceedings of the Conference on Computer Vision, Graphics, and Image Processing, Taipei, Taiwan, 24–27 June 2017.
- Chang, C.-Y.; Srinivasan, K.; Wang, W.-C.; Ganapathy, G.P.; Vincent, D.R.; Deepa, N. Quality Assessment of Tire Shearography Images via Ensemble Hybrid Faster Region-Based ConvNets. *Electronics* **2020**, *9*, 45. [[CrossRef](#)]

6. Chang, C.-Y.; Wang, F.-C. Tire Bubble Defects Detection Using ResNet. In Proceedings of the National Conference on Web Intelligence and Applications, Yunlin, Taiwan, 18–19 October 2019.
7. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
8. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
9. Wang, Q.; Bi, S.; Sun, M.; Wang, Y.; Wang, D.; Yang, S. YOLOv3 Architecture. *PLoS ONE* **2019**. [[CrossRef](#)]
10. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 346–361.
11. Cai, Z.; Fan, Q.; Feris, R.S.; Vasconcelos, N. A unified multi-scale deep convolutional neural network for fast object detection. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016.
12. Zuiderveld, K. Contrast Limited Adaptive Histogram Equalization. In *Graphics Gems IV*; Elsevier: Amsterdam, The Netherlands, 1994; pp. 474–485.
13. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [[CrossRef](#)]
14. Lin, T.-Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.