



Article

Vision-Based System for Black Rubber Roller Surface Inspection

Thanh-Hung Nguyen ¹, Huu-Long Nguyen ¹, Ngoc-Tam Bui ², Trung-Hieu Bui ¹, Van-Ban Vu ¹,
Hoai-Nam Duong ¹ and Hong-Hai Hoang ^{1,*}

¹ School of Mechanical Engineering, Hanoi University of Science and Technology, Hanoi 100000, Vietnam; hung.nguyenthanh@hust.edu.vn (T.-H.N.); long.nguyenhuu@hust.edu.vn (H.-L.N.); hieu.bt195010@sis.hust.edu.vn (T.-H.B.); ban.vv194910@sis.hust.edu.vn (V.-B.V.); nam.dh195107@sis.hust.edu.vn (H.-N.D.)

² Shibaura Institute of Technology, Tokyo 135-8548, Japan; tambn@shibaura-it.ac.jp

* Correspondence: hai.hoanghong@hust.edu.vn; Tel.: +84-934-493-466

Abstract: This paper proposes a machine vision system for the surface inspection of black rubber rollers in manufacturing processes. The system aims to enhance the surface quality of the rollers by detecting and classifying defects. A lighting system is installed to highlight surface defects. Two algorithms are proposed for defect detection: a traditional-based method and a deep learning-based method. The former is fast but limited to surface defect detection, while the latter is slower but capable of detecting and classifying defects. The accuracy of the algorithms is verified through experiments, with the traditional-based method achieving near-perfect accuracy of approximately 98% for defect detection, and the deep learning-based method achieving an accuracy of approximately 95.2% for defect detection and 96% for defect classification. The proposed machine vision system can significantly improve the surface inspection of black rubber rollers, thereby ensuring high-quality production.

Keywords: rubber roller; surface inspection; machine vision; deep learning; classification



Citation: Nguyen, T.-H.; Nguyen, H.-L.; Bui, N.-T.; Bui, T.-H.; Vu, V.-B.; Duong, H.-N.; Hoang, H.-H. Vision-Based System for Black Rubber Roller Surface Inspection. *Appl. Sci.* **2023**, *13*, 8999. <https://doi.org/10.3390/app13158999>

Academic Editor: Silvia Liberata Ullo

Received: 21 June 2023

Revised: 1 August 2023

Accepted: 2 August 2023

Published: 6 August 2023



Copyright: © 2023 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/>).

1. Introduction

Rubber rollers are vital components of machines that consist of an inner tube or shaft coated with elastomer compounds. They possess numerous desirable properties, including impact strength, shock absorption, compression and deflection, abrasion and chemical resistance, high coefficient of friction, and controllable degree of hardness, making them highly advantageous. They are an ideal choice for handling manufactured goods, causing no harm to the item or the roller itself, unlike metal rollers. In addition, rubber rollers' repair and re-coating are less time-consuming and require less investment than repairing a metal core, thereby extending their service life. They are commonly used in applications that require high surface durability with low to medium hardness, and their proper design and engineering of rubber compounds allow them to withstand mechanical and thermal stresses that may lead to degradation.

The primary focus of this research paper is on black rubber rollers, which have extensive use in printers and copiers. These rollers play a vital role in the printing process as they help to transfer ink to paper, and any surface defect can significantly impact the output's quality. As a result, inspection of these rollers is crucial to avoid issues such as heavy downtime, frequent roller replacement, plate wear, and printed material stage.

Surface defects on rubber rollers include scarring and damage from mechanical collisions, aging-related corrosion, and lack of material, among others. The rollers' cylindrical surface, chamfers, and end surfaces frequently contain these flaws. The defects are categorized into ten main categories, which include damage, corrosion, material lacking at the

chamfer, grind lacking, scratches, porosity, flash defects, reverse rubber, smaller size rubber, and no rubber rollers, as illustrated in Figure 1.

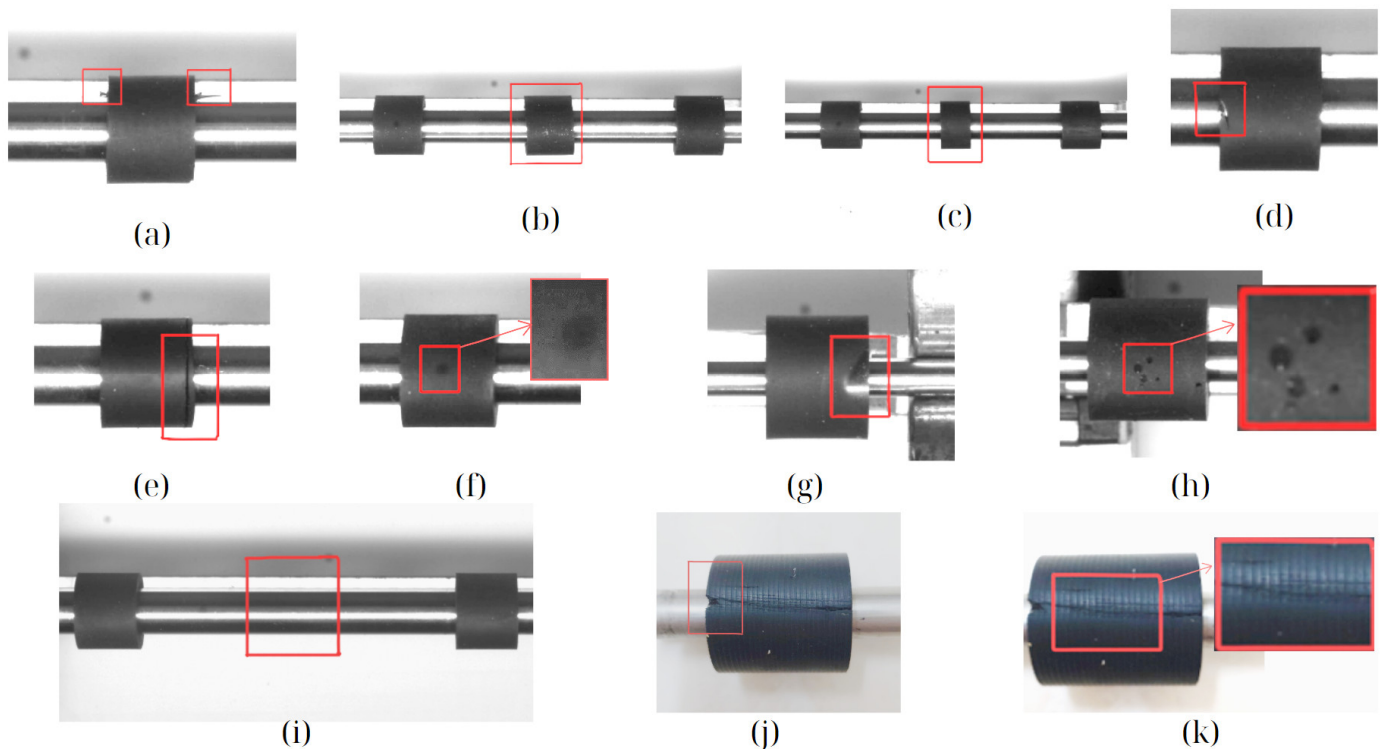


Figure 1. Typical defects on black rubber rollers: (a) flash defects; (b) reverse rubber; (c) smaller size rubber; (d,e) damage; (f) corrosion; (g) lack of grinding; (h) porosity; (i) no rubber rollers; (j) lack of material at the chamfer; and (k) scratches.

Figure 1 provides images of the cylindrical surface of a rubber roller, which highlight the importance of detecting surface defects to ensure the roller's performance, stability, productivity, and product quality. Visual inspection is a common method for detecting defects using either the raw human eye or inspection equipment or both these methods. With the development of computers and highly sensitive cameras, visual inspection uses digital images and deep learning to quickly and consistently identify defects in the manufacturing line such as in rubber tire, which is a flexible composite structure composed of rubber and was inspected by [1–4]. These papers have proposed tire defect classification based on different convolutional networks, which prove the robustness and effectiveness of the improved networks with reduced tire defect detection time. Although visual inspection is a useful tool for detecting defects, it presents several challenges when applied to black rubber rollers. In contrast to various inspection scenarios such as circuit solder inspection, plastic bottle defect detection, and metal product surface defect detection, manual inspection still predominates in the actual production process, leading to low inspection efficiency and accuracy.

The conventional inspection methods using light and magnifying glasses and rely on visual analysis, such as edge and line detection, often encounter challenges in accurately identifying and categorizing each type of defect on a rubber roller surface. The reason for this is difficulty in extracting the features of these defects. Typically, defects are treated as arbitrary targets, and in order to highlight them, a difference in reflection of the foreground and the background is needed. However, this method fails to utilize the unique characteristics of each defect, which can lead to the misclassification of certain synthetic marks like oil stains as defects. As a result, this approach can lower accuracy and lead to a low recall rate. Nevertheless, these limitations can be substantially mitigated by utilizing the appropriate lighting system.

This research paper presents a machine vision system which has been designed to inspect the surface of black rubber rollers. The system is capable of identifying and localizing major categories of faults on the outer parts of black rubber rollers and can determine whether a black rubber roller meets quality standards, based on the areas and locations of the detected defects. To meet industrial requirements, we suggest a specialized lighting system for the inspection and detection of defects. In addition, we introduce two methods for detecting and classifying defects on the surface of black rubber rollers. The system is capable of replacing manual inspection and exhibits improved performance. This entire surface inspection system is valuable for industrial applications in black rubber roller defect detection and quality evaluation.

Vision-based systems have gained popularity in inspecting rubber roller surfaces due to their high accuracy and efficiency in defect detection. Numerous approaches have been proposed for detecting and classifying defects on black rubber roller surfaces. Zhu et al. [5] focus on defect detection in rubber rings. They have presented an edge detection algorithm based on digital image processing. The algorithm involves preprocessing the target image to remove noise, utilizing Sobel operators to detect edges, and employing mathematical morphological algorithms for image dilation. The results show the algorithm's effectiveness in repairing fissures, enhancing image brightness, and achieving improved accuracy.

Bharathi et al. [6] have described their work on texture analysis for surface defect detection of rubber oil seals. Their approach includes extracting texture features from grayscale co-occurrence matrices with different spatial correlations. Since computing textural features for the entire image has proved ineffective due to local defect concentrations, the image is segmented prior to feature extraction. They also have proposed and implemented a unique preprocessing method.

Similarly, Meng et al. [7] have presented a special algorithm for detecting surface defects on rubber hose surfaces using computer vision technology and the HALCON algorithm. Their method includes defect classification to reduce misjudgment and improve the accuracy of the visual system, resulting in accurate defect detection.

Ho et al. [8] have developed an automated optical inspection system for silicone rubber gaskets, utilizing traditional rule-based and deep learning detection techniques. The system aims to detect characteristic defects in gaskets created during the manufacturing process, achieving high accuracy using convolutional neural networks (CNNs) and advanced image capturing and generating approaches.

He et al. [9] have addressed the challenges of identifying surface defects on oil seals using a visual detection method (VDM). They have proposed an approach of segmenting an image into regions, using gray level changes in the radial direction on the outer part of the oil seal. Additionally, they have introduced a circumferential background difference algorithm considering reflection inequality and low contrast, achieving high recall and precision rates in defect detection.

Jiamin Tao et al. [10] have proposed a novel approach that combines the fringe projection technique with traditional visual inspection devices to overcome their limitations. Their method employs deep learning techniques and introduces the Padua Incremental Mask Labeling Method to accelerate the calibration process. They have designed a one-stage architecture deep learning network called YOLO-OurNet specifically for detecting defects on drum-shaped roller surfaces. Experimental tests have demonstrated impressive results, including reduced defect detection time, high accuracy rates, and improved object detection evaluation indices.

Shengping Wen et al. [11] have introduced a multi-task convolutional neural network that was created to find flaws in order to control the surface quality of bearing rollers. By using a shared convolutional neural network, the defects' characteristics were extracted, and the classification and positioning of the faults were determined simultaneously. Ruoxu Ren et al. [12] have presented a general method for automated surface inspection that only requires a small amount of training data. This method creates a classifier based on the attributes of picture patches that have been taken from a deep learning network that

has already been trained. Tian Wang et al. [13] have proposed an intricately crafted deep convolutional neural network (CNN) that can automatically extract potent features for defect detection with less prior knowledge about the images, while at the same time being noise-resistant. Chil-Chyuan Kuo et al. [14] have presented an optical inspection system for inspecting the bubbles in silicone rubber with high accuracy in the millimeter range. Rafia Nishat Toma et al. [15] have proposed a method to recognize and classify multiple rolling bearing faults. In this work, ensemble empirical mode decomposition (EEMD) is used to split the signal into various intrinsic mode functions (IMFs) in order to evaluate the fault characteristics. A continuous wavelet transform is then applied, transforming the 1D reconstructed vibration signal into a 2D image. The signal is moved into the time-frequency domain, which lessens the vibration signal's nonstationary effects. Additionally, Sier Deng et al. [16] have suggested an automatic detection system based on a machine vision technique in response to the high demand for productivity and bearing quality as well as the dearth of conventional detection methods.

In summary, these studies highlight the potential of vision-based systems in detecting and evaluating defects on rubber roller surfaces. Various approaches have been explored, ranging from traditional image processing techniques to deep learning-based methods. Each approach offers unique advantages and contributes to the advancement of defect detection in the field of rubber roller surfaces.

The paper is organized as follows: Section 2 details the design of the visual inspection system, encompassing both hardware and software systems. Sections 3 and 4 present the experimental results and discuss the developed machine vision system. Lastly, Section 5 summarizes the entire paper.

2. Visual Inspection System

2.1. Overview

Both the hardware system and software system are built to form the visual inspection system.

System hardware includes the desktop computer, shaft with black rubber rollers, V-mount, lighting system, mounting bracket, camera, and lens. The hardware system includes Intel®Core™ i7-7700HQ CPU @ 2.80 GHz (8 CPUs, ~2.8 GHz) with 8.00 GB RAM, NVIDIA GeForce GTX 1050, and Windows 10 is the operating system. The mechanical structure is shown in Figure 2 below. It mainly consists of the aforementioned hardware devices with the lighting system being ring lights or backlights.

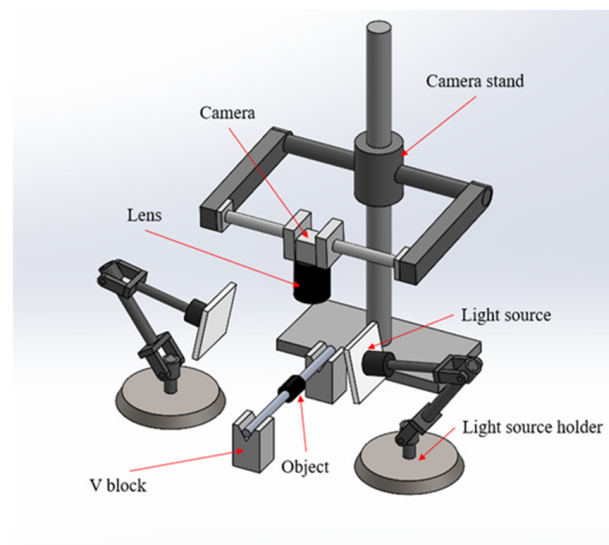


Figure 2. Mechanical structure.

The rubber roller has two heads and one cylinder; we only work on the cylinder, so we need a workplace and an industrial camera for image acquisition. When the axis is placed in the required position and fixed on the V-mount, the video acquisition function is activated. At work, a shaft containing four rollers rotates in place under the action of the mechanism. We use a Basler acA2040-90uc area scanning camera, manufactured by Basler, Germany, with a high resolution of 2040×2046 , to record the cylindrical surface of the roller. The choice of camera is considered according to the requirements of the distance to target and sharpness of the image.

There are strict lighting requirements for visual inspection, and the image's condition can be enhanced using the stable light. Selecting and setting up a constant light source is vital to highlighting the features of the defects. We set up two high-angle light sources. The two light sources are placed parallel to the axis of the roller. The method for choosing a lighting system is described in Section 2.2.

The software system is programmed in C++. The distance function is used as the ground for building the defect detection algorithm. Commonly used image processing algorithms, such as segmenting by level and morphological processing, were written with the help of OpenCV library. In addition, we also use a deep learning algorithm that uses the YOLO algorithm to automatically detect and classify errors and compare them with the traditional method used.

2.2. Illumination Design

In this lighting system, we use two backlights, one camera, one lens, and two V-mounts; the designed mounting system is shown in Figure 3 below.

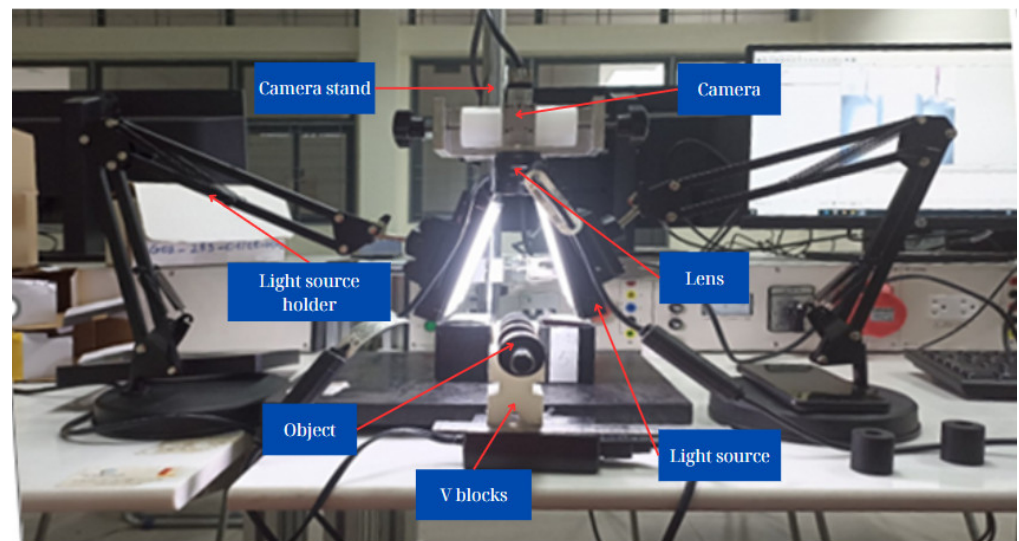


Figure 3. Lighting system.

Two backlights placed symmetrically about the axis are tested with illumination angles from 0 to 90 degrees from the horizon. After a series of tests, it was discovered that the 60-degree angle gives the best image quality, and the defects are most visible. The distances from the center of the shaft to the lens and from the lamp to the shaft are shown in Figure 4 below. These distances are a function of the size of the rubber roller as well as the size of the backlights.

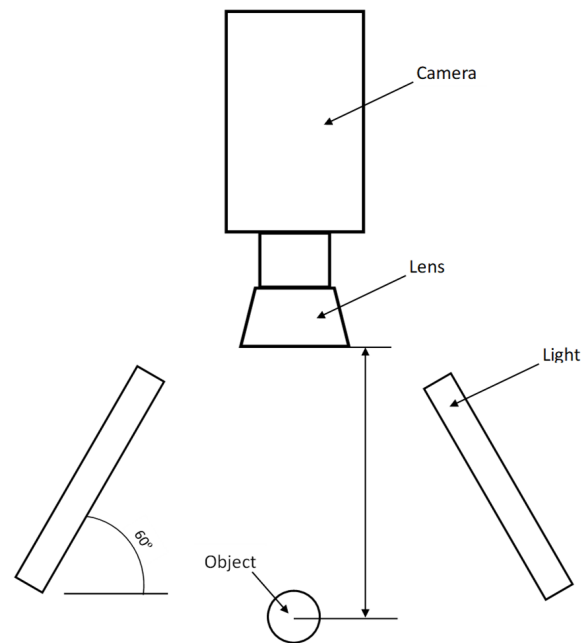


Figure 4. Lighting scheme.

2.3. Defect Detection Based on Proposed Traditional Method

Rubber roller defect images only provide useful information about the defect area, typically tiny in comparison with the overall rubber region, especially if the rubber surface condition is fine. Therefore, a large portion of the image is considered background and needs to be removed for efficiency. However, in most cases interference from the imaging environment introduces a large amount of noise, making it difficult to distinguish important image regions from unwanted regions. As a result, it is necessary to develop image preprocessing operations to minimize environmental disturbances. OpenCV is utilized to make the process as fast and efficient as could be expected. OpenCV implements many computer vision algorithms and is written in the C programming language. Image preprocessing is done in several steps, which are described below.

Since the visible area of the camera is wider than the working area, we first extract the working area from the video obtained during image acquisition. The extracted area will be further processed in the image preprocessing stage. The traditional defect detection process is depicted in Figure 5 below.

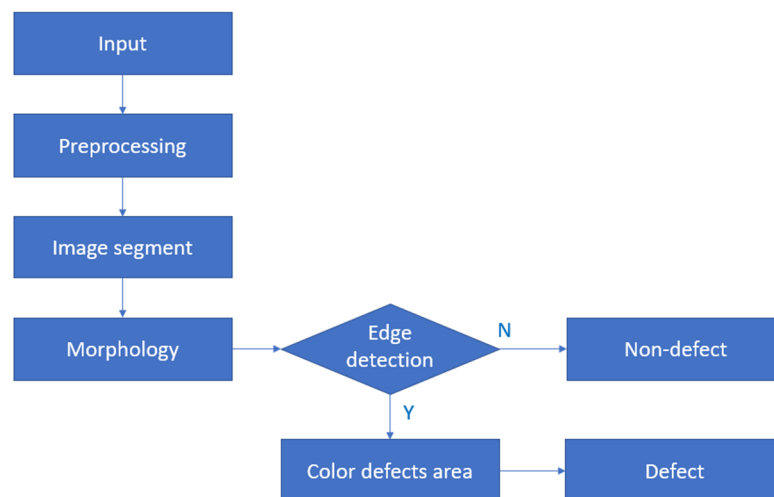


Figure 5. Proposed traditional method.

After extracting the ROI region from the input video, images extracted from the ROI region are put into preprocessing. We use the same input data set for both methods, and an image can include more than one defect, as shown in the table below. Then, the thresholding stage highlights the error on the video background using black and white values. Morphology and edge detection make defects clearer and easier to see. If edge detection returns a value other than zero, the defect area will be colored, and the rubber roller with defects can be detected, and vice versa, if the rubber roller is in good condition.

2.3.1. Power-Law Transformation

The power law (gamma) transformation function is generally described as [13]:

$$s = c * r^\gamma \quad (1)$$

The output and input pixel values are denoted using s and r , respectively, and the positive constants are denoted using c and γ . Power-law curves with $\gamma < 1$ map a narrow range of dark input values into a wider range of output values, whereas higher input values map the opposite. This is similar to the log transformation. In a similar vein, the opposite occurs when $\gamma > 1$.

Gamma correction, gamma encoding, and gamma compression are all other names for this. The following curves are produced by normalizing the r values from 0 to 1. The scaling constant c that corresponds to the bit size used then multiplies them. With this transformation, the contrast of the picture will be enhanced, and the defects will be shown more clearly.

Then, the image is converted to a new negative gray scale (shown in Figure 6). This facilitates the processing and viewing of processed images. All preprocessing steps are explained below.

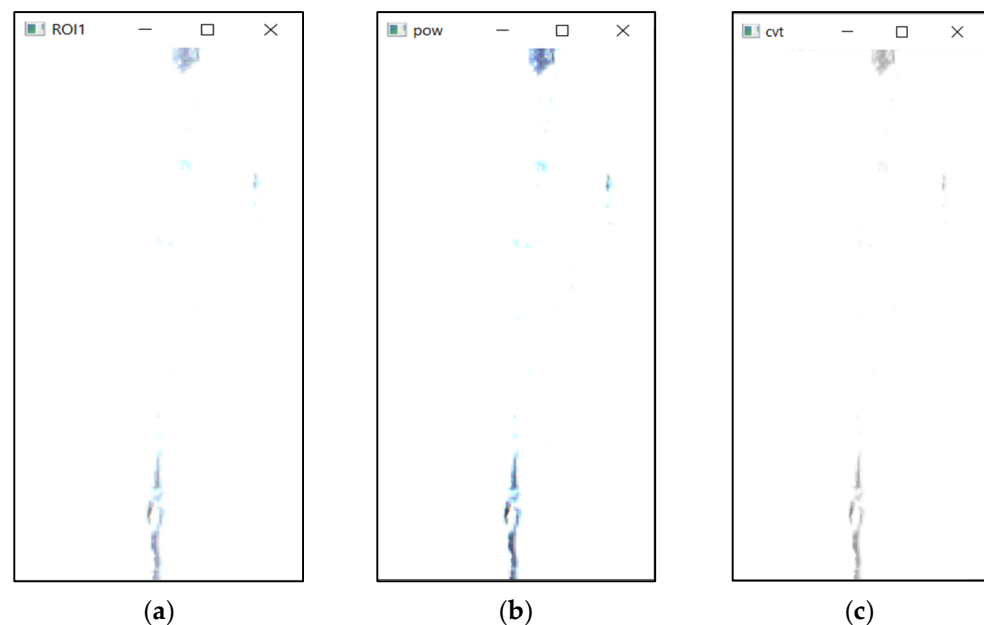


Figure 6. (a) ROI (original image); (b) result of applying the power-law transformation; (c) gray scale image.

2.3.2. Laplacian Filters

A Laplacian filter or a Laplacian edge detector is used to compute the second derivatives of an image, measuring the rate of change of the first derivatives. This determines whether changing adjacent pixel values are from an edge or a continuous progression.

The Laplacian is often applied to an image that has first been smoothed with something approximating a Gaussian smoothing filter in order to reduce its sensitivity to noise. One

of the main issues that could affect the results is random noise. All of the images may be affected by random and uniform noise due to the physical characteristics of rubber rollers, which must be removed. By using the Laplacian filter, the photos are smoother, and the sensitivity to noise is also significantly reduced.

The Laplacian $L(x,y)$ of an image with pixel intensity values $I(x,y)$ is given by [13]:

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad (2)$$

This can be calculated using a convolution filter. Since the input image is represented as a set of discrete pixels, we have to find a discrete convolution kernel that can approximate the second derivatives in the definition of the Laplacian. Two commonly used small kernels are shown in Figure 7.

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	1	-1	-1

Figure 7. Kernel matrix.

Using one of these kernels, the Laplacian can be calculated using standard convolution methods. The result of applying the Laplacian is shown in Figure 8 below.

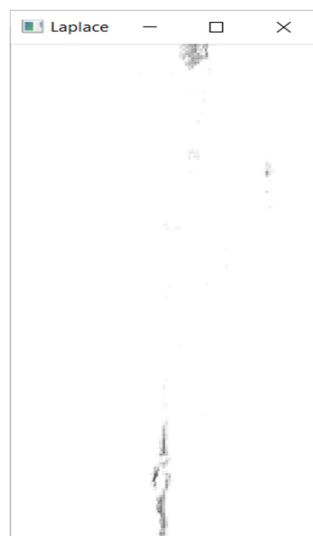


Figure 8. As a result of applying the Laplacian, the image is smoother.

2.3.3. Image Segmentation: Thresholding

Image segmentation is an image processing method that creates a binary image based on thresholding the pixel intensity of the original image. It is usually used on grayscale images but can be used on color images as well. The image intensity threshold (the relative brightness of the image) is either manually set to a specific value or set automatically by the application. Pixels below the threshold will be black (bit value 0), and pixels above the threshold will be white (bit value 1). The thresholding process is sometimes described as dividing an image into foreground (black) values and background (white) values.

Contrast-based simple thresholding operations set a single global threshold value for all pixels in an image, regardless of any differences in local contrast. Adaptive thresholding, a type of more advanced thresholding, generates a threshold value by sampling

progressively larger numbers of image regions. When performing OCR on images, threshold quality is especially important. Materials that are foxed, mottled, stained, or faded irregularly can make it hard to distinguish the foreground (text) from the background. The defects will be identified via thresholding as background values (white) and foreground values (black). The result of applying thresholding is shown in Figure 9 below.

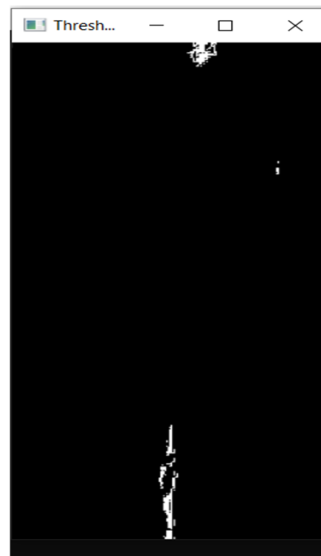


Figure 9. Result of applying thresholding.

2.3.4. Morphological Image Processing

Mathematical morphology and set theory are the foundations of morphological filters. There are many different kinds of morphological filtering, but after analyzing the results, erosion and dilation filters are the best ones in the experiments. First, we use an erosion filter to remove small noises or insignificant defects. Then, the dilation filter assists with filling minor holes in the picture, making the main defect seamless and more detailed. Along with that, the object also returns to its original size after being scaled down by the erosion filter.

Figure 10 points the outcome of using an erosion and dilation morphological filter to the defect image with previously applied thresholding.



Figure 10. The defect is clearly defined and now stands out as a result of using the morphological operator.

2.3.5. Edge detection

The first derivative of the image intensity is one of the most widely used methods for detecting edges because it has a constant value throughout the transition and is zero in all regions with low variance. Therefore, the change in intensity (potential edge) is revealed as an abrupt first derivative change. This feature is usually used for edge detection. The Canny algorithm is based on this technique.

The Canny algorithm includes three main steps:

- Find the intensity gradient of the image: in this step, the scale of the gradient vector is calculated for each pixel.
- Non-maximum suppression: the aim of this step is to 'thin' the edge to obtain a one-pixel width edge.
- Threshold hysteresis: finally, a two-step threshold hysteresis is applied in order to decrease the fake edges.

After applying the Canny algorithm, the contours of defects are detected in order to proceed to coloring the error area in the following step. A contour search function is set up to merge small contours that are closely spaced (according to a predetermined weight) into one large contour; then, coloring the error area is easy, and the error is visible and displayed most clearly. After this step, the errors are identified completely correctly as shown in Figure 11.

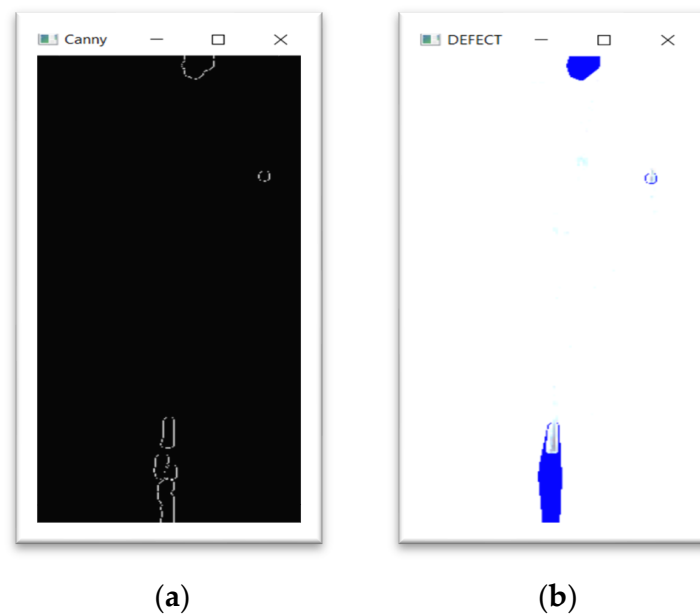


Figure 11. (a) As a result of applying the Canny edge detector algorithm, the defect is now clearly identified. (b) The defects are now clearly identified by coloring blue color.

2.4. Defect Detection Based on the Deep Learning Method

In addition to using the traditional method, we propose using an artificial intelligence network to identify errors on the rubber roller.

In order to be able to accurately and quickly detect errors with a short delay, and with the requirement to quickly process photos and videos at hundreds of frames per second, we use the YOLOv7 network with advanced techniques. Such as with CNNs, skip connection and multi-scale training are used to improve error detection accuracy and speed. Figure 12 below shows an overview block diagram of the implementation of defect detection using deep learning.

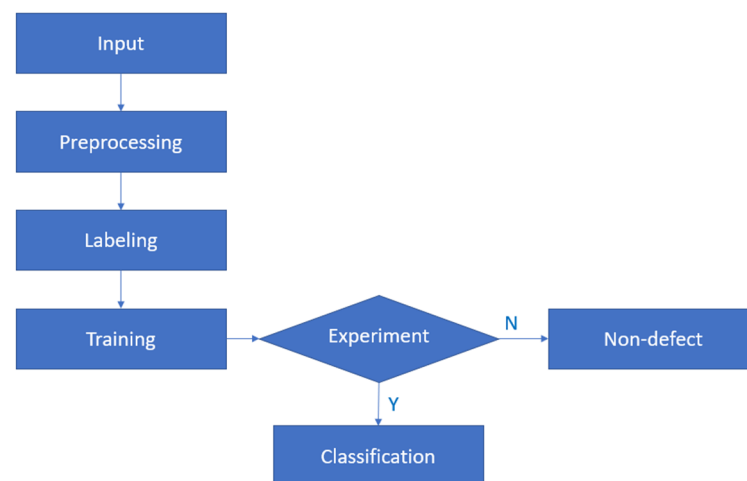


Figure 12. Defect detection using deep learning.

After collecting data (videos) from the machine vision system presented in Section 2, we proceed to separate images from video; then, the input data set has been established. Because the image quality is not good enough to meet the accuracy and performance requirements of the defect detection process, data preprocessing is applied. The preprocessed images are labeled in the next stage. Four typical defect types are selected for labeling. After that, the data training process is conducted on Google collapse. At the end of the process is the experimental result, and the rubber rollers are classified according to each type of defect.

Before going to the network, the input image (resolution) is resized to 640×640 , which then is used as the input to the model. The entire network architecture after the fusion is shown in Figure 13. The backbone network is first used to extract features from the input images. In order to maintain multi-scale information, the neck network receives multi-scale feature maps from the backbone network, which provides a variety of scales and serves as the neck network's input. After the neck network, shallow fine-grained information and deep semantic information from feature maps are fused, increasing the expressive capacity of the network and assigning the multi-scale learning task to numerous detection networks of various sizes. The output of the detection prediction is created after the feature information has been combined and modified.

Images' textures, colors, and shapes are extracted as features using a backbone network. It may offer a variety of scales, sensing field sizes, and center steps, accommodating the needs of different scales and categories. Figure 13A illustrates the backbone network's feature extraction process. First, the backbone network passes through four CBS modules for convolution, normalization, and activation. Next, the E-ELAN block and MP block are used to extract features in an alternating fashion. Finally, the result of the final three E-ELAN blocks is used as the income data for the neck. As illustrated in Figure 14a,e, the MP block is made up of MaxPool and CBS modules, and the E-ELAN block is made up of several convolutional layers.

The neck network's job is to fuse the learned multi-scale information together and distribute the backbone network's multi-scale output learning to multiple feature maps. This improves the model's perceptual wildness, successfully separates the most important contextual features, and partially avoids the image distortion problem. As shown in Figure 13C, the model's prediction results are obtained by combining the feature data with the 1×1 convolution to produce the final prediction data. This is done after the backbone and neck networks have extracted two features from the input image.

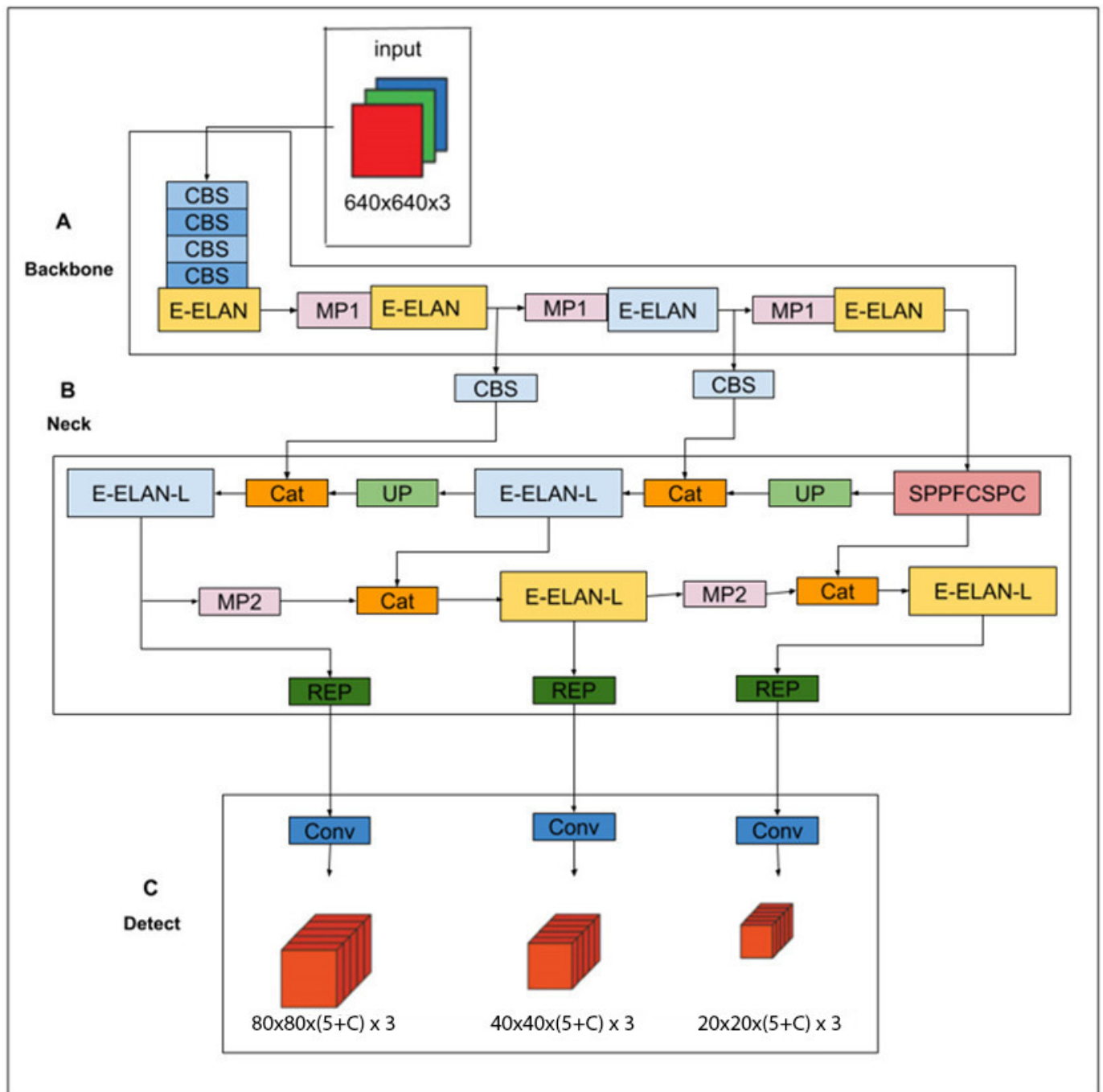


Figure 13. The whole model structure. (A) Backbone network structure for extracting feature; (B) neck network structure for fusion feature; (C) detect network structure for obtaining the inference output, and C is the number of predefined classes.

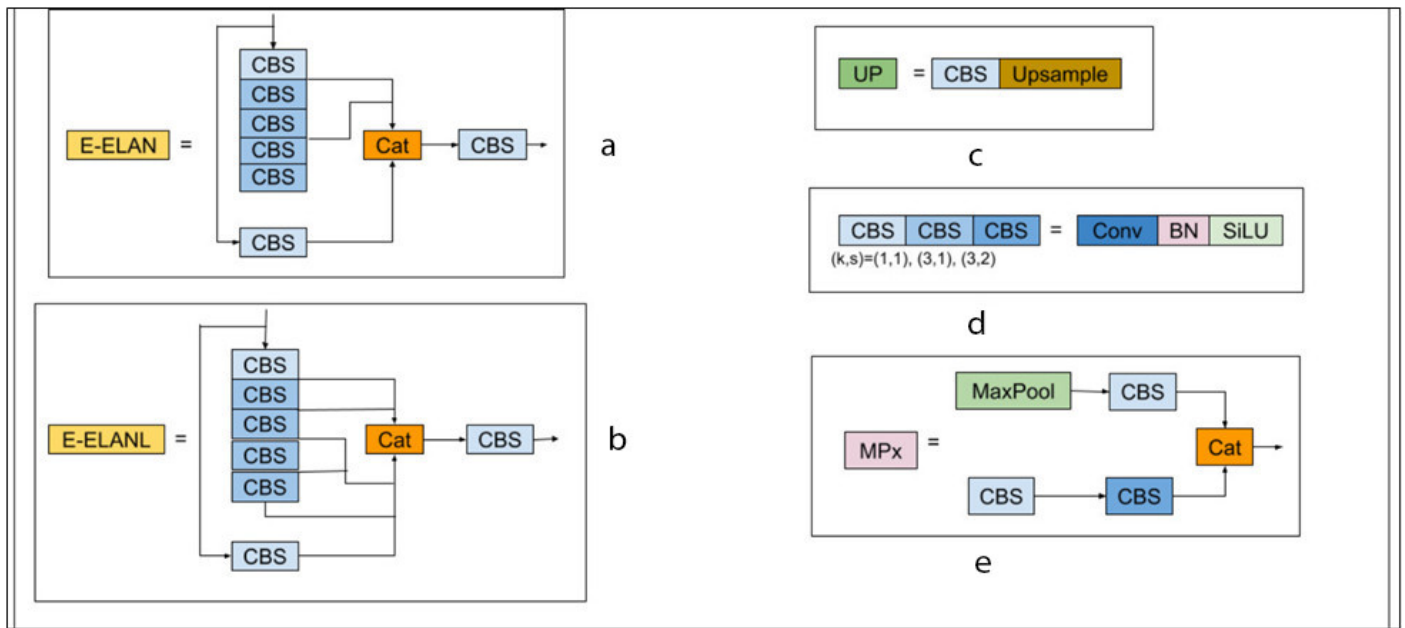


Figure 14. Image of the model’s blocks. (a) The ground structure of the E-ELAN block; (b) ground structure of the E-ELAN-L block; (c) upsampling block; (d) layout of different convolution blocks, where k denotes the convolution kernel size and s denotes the step size; (e) basic structure of the MP module.

2.4.1. Preprocessing

A total of 1063 defective pictures of the rubber roller are used for detection. Images are taken from video obtained through the lighting system. Detected faults include no defects, torn areas, porosity, and damage. To improve the accuracy during training, we perform preprocessing of these images.

First, gamma correction is used to increase the contrast of the input image. The gamma factor is adjusted to achieve the most suitable image quality, the higher the gamma factor, the higher the contrast of the image. After some experimentation, the gamma factor value is chosen to be 6.

$$\text{img_contrast} = ((\text{image}/255.0)^\wedge \text{gamma}) * 255. \tag{3}$$

Then, image quality is improved using the GaussianBlur filter. The resulting image will then be less noisy and smoother. Filter parameters are adjusted to obtain the best image quality and are written using OpenCV command `blur = cv2.GaussianBlur(img_contrast, (3, 3), 0)`.

2.4.2. Dataset and Labeling

Our dataset is collected from images after preprocessing. After training the model, we test it on a test dataset including 230 images. The training set includes 850 images for the training process. The validation set includes 213 images to evaluate the performance of the model during training. The validation set helps prevent over-fitting to help achieve the best results.

We perform labeling and identify three defects on the rubber roller surface: damage, torn areas, and porosity. For images having no defects on the surface, nothing will be displayed.

Figure 15 shows the results of the defect detection. The purple box is associated with damage, the green box is associated with tears, and the blue box is associated with porosity. The defect’s probability and category are shown either above or below the box.

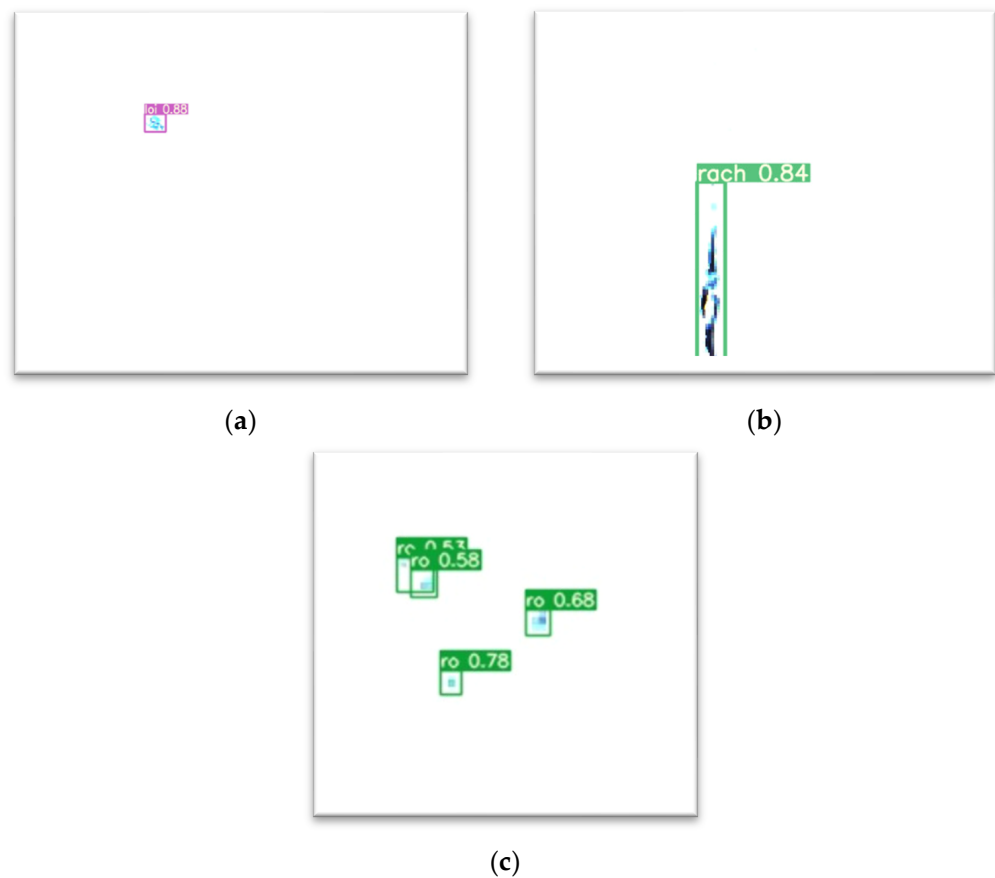


Figure 15. Defect detection results: (a) damage; (b) torn areas; (c) porosity. The purple box is associated with damage, the green box is associated with tears, and the blue box is associated with porosity.

2.4.3. Training and Accuracy

The training parameters of the training process used in the experiment are shown in Table 1.

Table 1. Training parameters.

Parameter	Value	Parameter	Value
Learning rate	0.01	Weight decay	0.005
Batch size	12	Momentum	0.937
Image size	640 × 640 × 3	Epochs	150

2.4.4. Evaluation Metrics

The evaluation indices used to evaluate the performance of the algorithm in this paper are precision (P), recall (R), mean precision (mAP), and $F1$ score.

Precision expresses the proportion of positive samples in the samples with positive prediction results. It is calculated as follows [13]:

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

Recall expresses the prediction result as the proportion of the actual positive samples in the positive samples to the positive samples in the whole sample, calculated as follows [13]:

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

The F1 score is the weighted average of precision and recall, calculated as follows [13]:

$$F1 = \left(\frac{2}{Recall^{-1} + Precision^{-1}} \right) = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{6}$$

The model’s ability to distinguish between positive and negative samples is shown by its accuracy. The model’s ability to distinguish between negative samples increases with accuracy. The model’s capacity to identify positive samples is reflected in recall. The model’s capacity to identify positive samples increases with recall. The two are combined in the F1 score. A stronger model is indicated by a higher F1 score.

Average accuracy (AP) is the average value of the highest precision under different recall conditions. (Usually, AP is calculated for each category separately.)

The mean average precision (mAP) is the mean value of the average precision and the mean AP value of each category. The calculation formula is as follows [13]:

$$mAP = \sum_{j=1}^s \frac{AP(j)}{S} \tag{7}$$

where S is the number of all categories, and the denominator is the sum of the APs of all categories. The object detection object in this study comprises three defects of the rubber roller; therefore, S = 3.

3. Results

3.1. Defect Detection Results

3.1.1. Traditional Method

We use 230 images of rubber roller defects using the lighting system described in Section 2, in which there are 197 images with defects, including damage, torn areas, and porosity, and 33 images without defects as shown in Table 2. After applying the traditional method, the defect areas are shown quite clearly, and the accuracy is approximately 98%. However, this method only gives results in defect and non-defect, unclassified defect types. The results are shown in Table 3 below.

Table 2. Input data set.

Defect Categories (Images)	Damage	Torn	Porosity	Non-Defect	Total
Test set	173	56	16	33	230

Table 3. Results of traditional method.

Type	TP (Images)	FP (Images)	Precision (%)
Defect	195	2	98.98
Non-defect	32	1	96.97
Image size	227	3	98.70

3.1.2. Deep Learning Method

The training parameters and graphs of the training process used in the experiment are shown in Figure 16 below.

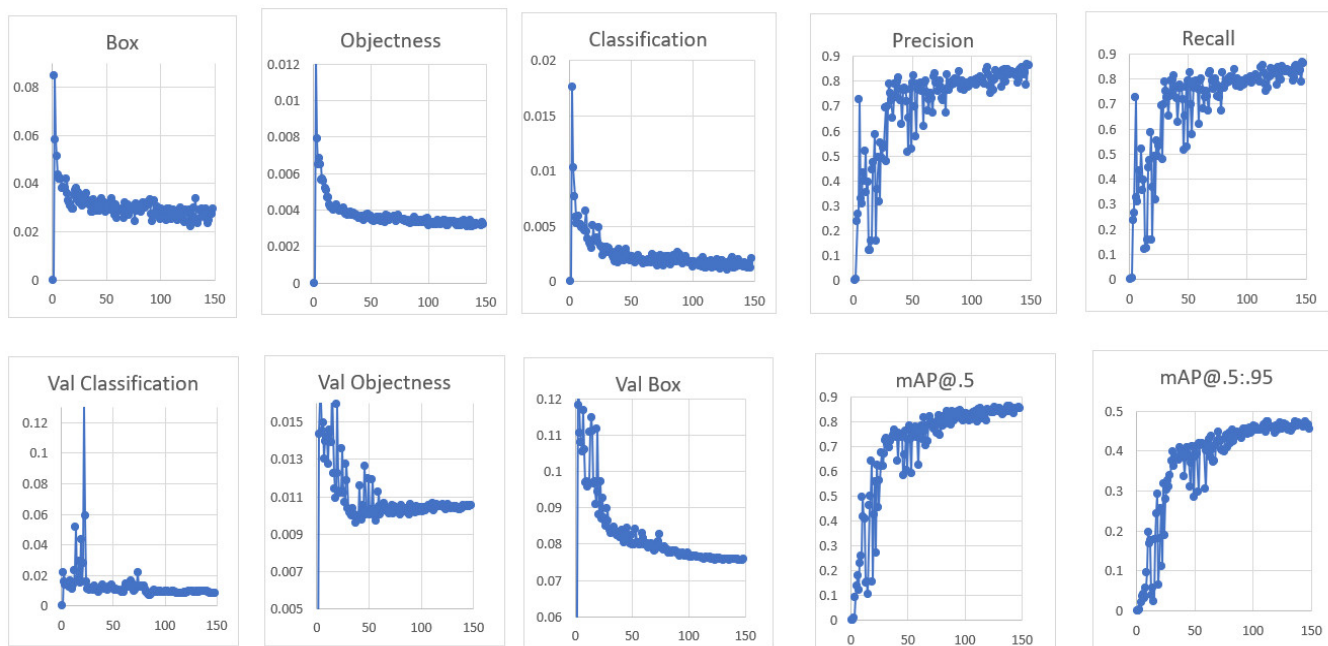


Figure 16. Training parameters and graphs. The lower the Box, Objectness, Classification, Val Box, Val Objectness, and Val Classification parameters and the higher the parameters P , R , $mAP@.5$, and $mAP@.5:.95$, the better performance the model will achieve, with good object recognition ability.

The performance index values of the model are shown in Table 4.

Table 4. Performance index values.

	P	R	$F1$	$mAP@.5$	$mAP@.5:.95$	PR
Val	0.95	0.95	0.83	0.85	0.46	0.85

The test set consists of 230 images, each of which may contain multiple defects. The detail is shown in Table 2. After applying the deep learning method to identify defects, the model correctly recognized 278 defects, and the remaining 14 defects were incorrectly identified; the accuracy reached 95.21%.

3.2. Defect Classification Results

In addition to being able to distinguish whether the quality of the rubber roller is up to standard due to a general fault or not, the deep learning method can also classify each type of defect present on the rubber roller, which the traditional method is not capable of.

The outcome of the system evaluation is shown in Table 5. Successfully classified images are displayed in the Success column, while incorrectly classified images are displayed in the Error column. In this table, an image that has a type of defect but has been classified as another type of defect or as a non-defect image will be considered a wrongly classified image. The success rate is greater than 93% for each type of defect in this category, and in most cases, it reaches 100%. The success rate is greater than 95% on average.

Table 5. Results of deep learning method.

Type	TP (Images)	FP (Image)	Precision (%)
Damage	163	10	94.22
Torn areas	54	2	96.43
Porosity	15	1	93.75
Non-defect	33	0	100
	265	13	95.32

4. Discussion

4.1. Comparing the Two Methods

Edge detection, segmentation, and line detection, which are common manufacturing techniques, are unable to accurately extract the internal structure of any defect. Normally, the damage is treated as a random target, and the difference in reflectivity between the target and the background is used for detection. The rubber roller's qualification is then evaluated based on the position and area of the target. However, this approach often leads to misidentification of textures, stains, oil stains, etc., as defects. This results in low accuracy and a reduced recall rate during the detection process. In some cases, we need to know how many different kinds of defects there are and how often they happen, so we can adjust the production process accordingly. Conventional manufacturing surface inspection techniques cannot accomplish this. Deep learning has emerged to make up for the drawbacks of traditional algorithms. Since deep learning algorithms have shown better performance in object recognition and classification tasks, deep neural networks can be used to detect different types of defects and identify similarities between them. Accurate fault classification can be obtained from large amounts of data.

After applying two methods, we find that each method has its own advantages and disadvantages, and both methods give quite good defect identification results and can be applied in industry.

- Traditional method

This method does not require multiple input images to train the model and can be widely applied in manufacturing industries to detect defects, with low cost and good defect detection ability. However, its accuracy is highly dependent on the lighting system. Therefore, if the lighting system changes, the accuracy of the detection may also change. This makes it difficult to identify specific errors using this method.

- YOLOv7

The deep learning method is able to classify specific types of defects on the rubber surface such as damage, erosion, etc.; multiple input images can be used to increase accuracy, and it can be applied in rubber product visual inspection systems to classify defects and improve reliability during inspection.

However, it is necessary to use many input images to train the model, and a certain level of deep learning knowledge is required.

4.2. Other Illumination Designs

In order to establish the best lighting system, we have conducted experiments on many different lighting systems. Through our research, we have found that lighting systems such as the dome lighting system and backlight lighting system are not suitable for identifying defects on a black surface. Therefore, we conduct experiments on lighting systems with more potential such as the darkfield lighting system, a lighting system using ring light, and a spotlight lighting system using two lights on both sides. The lighting systems are set up as shown in Figure 17.

After setting up the lighting systems, we proceeded to collect images for comparison, and the results are shown in Figure 18 below. Looking at the results, it can be seen that the spotlight lighting system using two lights on both sides gives the best display results, the widest working area (white area), and the sharpest display defects.

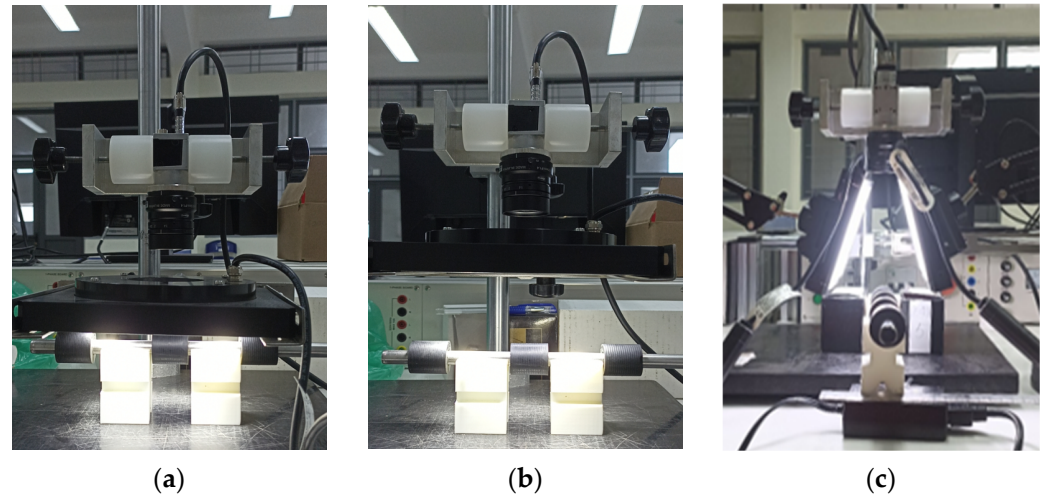


Figure 17. (a) Darkfield lighting system, light source close to the object; (b) lighting system using ring light, light source close to the lens; (c) spotlight system, light sources arranged diagonally on both sides of the object.

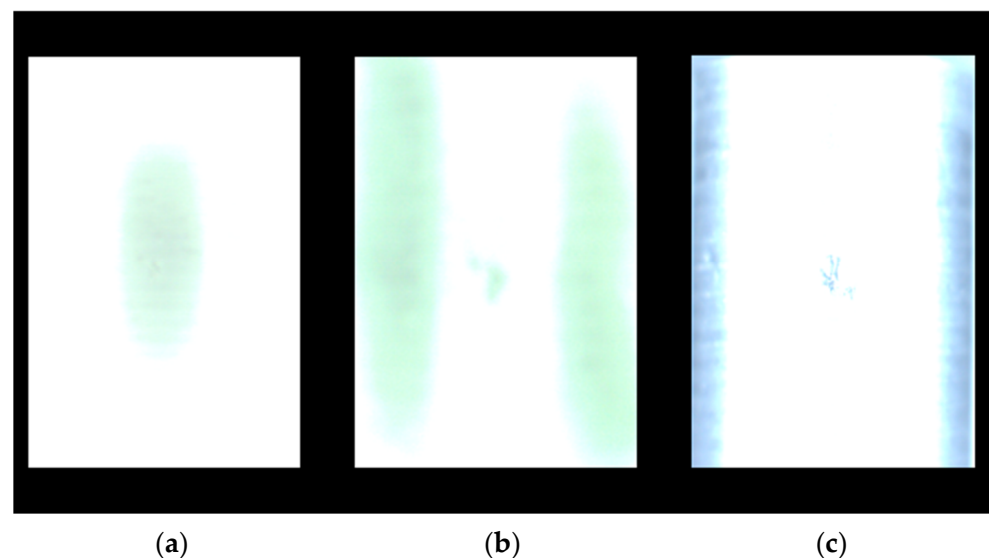


Figure 18. Image collecting: (a) darkfield lighting system: defects are not displayed, small working area; (b) lighting system using ring light: defects are displayed but not sharp, the working area is still not optimal; (c) spotlight system: defects are displayed sharply, wide and optimal working area.

5. Conclusions

A machine vision system for surface inspection of rubber rollers is proposed in this paper. Traditional and a deep learning approaches are designed to identify flaws and compare them in order to control the product's quality. The defects are classified, and their positions are computed simultaneously using the deep learning technique with YOLOv7. Finally, we look at the position, category, and area of the defects to see if the rubber roller is qualified. Importantly, experiments are conducted to find the best lighting system. Both methods meet the requirements of industrial manufacturing, but the quantitative results demonstrate that our method is superior in terms of accuracy and robustness.

The traditional method cannot classify defects, and deep learning needs a lot of labeled data and depends on how well the hardware works. This is a limitation of our proposed method.

Traditional methods give better results to classify defects versus no defects than deep learning methods do. However, the deep learning method is capable of classifying defects with outstanding accuracy. A reasonable combination of both methods will help improve accuracy and classification ability as well as optimize processing time.

We will continue to improve the algorithm and network structure in order to lower their computational costs and enable their truly widespread application in industrial manufacturing.

Author Contributions: Supervision: H.-H.H.; conceptualization: T.-H.N.; methodology: T.-H.N. and H.-H.H.; validation: N.-T.B. and H.-L.N.; investigation: T.-H.B., V.-B.V., and H.-N.D.; resources: T.-H.N. and H.-H.H.; writing—original draft preparation: H.-L.N. and T.-H.N.; writing—review and editing: H.-H.H. and N.-T.B.; project administration: H.-H.H.; funding acquisition: N.-T.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work was also supported by the Centennial Shibaura Institute of Technology Action to mark the 100th anniversary of Shibaura Institute of Technology and entry into the top ten Asian Institutes of Technology list and Support for Autonomous Higher Education Project (SAHEP) funded by World Bank for Hanoi University of Science and Technology.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: MDPI Research Data Policies.

Conflicts of Interest: The authors declare that there are no conflict of interest.

References

1. Wang, R.; Guo, Q.; Lu, S.; Zhang, C. Tire defect detection using fully convolutional network. *IEEE Access* **2019**, *7*, 43502–43510. [[CrossRef](#)]
2. Zheng, Z.; Shen, J.; Shao, Y.; Zhang, J.; Tian, C.; Yu, B.; Zhang, Y. Tire defect classification using a deep convolutional sparse-coding network. *Meas. Sci. Technol.* **2021**, *32*, 055401. [[CrossRef](#)]
3. Yang, S.; Jiao, D.; Wang, T.; He, Y. Tire speckle interference bubble defect detection based on improved faster RCNN-FPN. *Sensors* **2022**, *22*, 3907. [[CrossRef](#)] [[PubMed](#)]
4. Lin, S.-L. Research on tire crack detection using image deep learning method. *Sci. Rep.* **2023**, *13*, 8027. [[CrossRef](#)] [[PubMed](#)]
5. Zhu, H.-L.; Wang, X.-R.; Zhu, H.-L. Research on Defect Detection in Rubber Rings. In *Proceedings of the ICCEAE2012, AISC*; Springer: Berlin/Heidelberg, Germany, 2013; Volume 181, pp. 69–74.
6. Bharathi, S.S.; Radhakrishnan, N.; Priya, L. Surface Defect Detection of Rubber Oil Seals Based on Texture Analysis. In *Lecture Notes in Electrical Engineering, Proceedings of the Fourth International Conference on Signal and Image Processing 2012*; Springer: New Delhi, India, 2013; Volume 222, pp. 207–216.
7. Meng, F.; Ren, J.; Wang, Q.; Zhang, T. Rubber hose surface defect detection system based on machine vision. *IOP Conf. Ser. Earth Environ. Sci.* **2018**, *108*, 022057. [[CrossRef](#)]
8. Ho, C.-C.; Su, E.; Li, P.-C.; Bolger, M.J.; Pan, H.-N. Machine Vision and Deep Learning Based Rubber Gasket Defect Detection. *Adv. Technol. Innov.* **2020**, *5*, 76–83. [[CrossRef](#)]
9. He, Z.; Liu, J.; Jiang, L.; Zhao, S.; Zhang, L.; Cui, L. Oil seal surface defect detection using superpixel segmentation and circumferential difference. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 1729881420976511. [[CrossRef](#)]
10. Tao, J.; Zhu, Y.; Jiang, F.; Liu, H.; Liu, H. Rolling Surface Defect Inspection for Drum-Shaped Rollers Based on Deep Learning. *IEEE Sens. J.* **2022**, *22*, 8693–8700. [[CrossRef](#)]
11. Wen, S.; Chen, Z.; Li, C. Vision-Based Surface Inspection System for Bearing Rollers Using Convolutional Neural Networks. *Appl. Sci.* **2018**, *8*, 2565. [[CrossRef](#)]
12. Ren, R.; Hung, T.; Tan, K.C. A Generic Deep-Learning-Based Approach for Automated Surface Inspection. *IEEE Trans. Cybern.* **2018**, *48*, 929–940. [[CrossRef](#)] [[PubMed](#)]
13. Wang, T.; Chen, Y.; Qiao, M.; Snoussi, H. A fast and robust convolutional neural network-based defect detection model in product quality control. *Int. J. Adv. Manuf. Technol.* **2018**, *94*, 3465–3471. [[CrossRef](#)]
14. Kuo, C.-C.; Chen, Y.-R. Rapid optical inspection of bubbles in the silicone rubber. *Optik* **2013**, *124*, 1480–1485. [[CrossRef](#)]

15. Nishat Toma, R.; Kim, C.-H.; Kim, J.-M. Bearing Fault Classification Using Ensemble Empirical Mode Decomposition and Convolutional Neural Network. *Electronics* **2021**, *10*, 1248. [[CrossRef](#)]
16. Deng, S.; Cai, W.; Xu, Q.; Liang, B. Defect detection of bearing surfaces based on machine vision technique. In Proceedings of the International Conference on Computer Application and System Modeling (ICCASM 2010), Taiyuan, China, 22–24 October 2010; pp. V4-548–V4-554. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.