sensors

MDPI

Article

Algorithms for Vision-Based Quality Control of Circularly Symmetric Components

Paolo Brambilla *, Chiara Conese, Davide Maria Fabris , Paolo Chiariotti and Marco Tarabini

Department of Mechanical Engineering, Politecnico di Milano, Via La Masa 1, 20156 Milan, Italy
* Correspondence: paolo1.brambilla@polimi.it

Abstract: Quality inspection in the industrial production field is experiencing a strong technological development that benefits from the combination of vision-based techniques with artificial intelligence algorithms. This paper initially addresses the problem of defect identification for circularly symmetric mechanical components, characterized by the presence of periodic elements. In the specific case of knurled washers, we compare the performances of a standard algorithm for the analysis of grey-scale image with a Deep Learning (DL) approach. The standard algorithm is based on the extraction of pseudo-signals derived from the conversion of the grey scale image of concentric annuli. In the DL approach, the component inspection is shifted from the entire sample to specific areas repeated along the object profile where the defect may occur. The standard algorithm provides better results in terms of accuracy and computational time with respect to the DL approach. Nevertheless, DL reaches accuracy higher than 99% when performance is evaluated targeting the identification of damaged teeth. The possibility of extending the methods and the results to other circularly symmetrical components is analyzed and discussed.

Keywords: vision-based quality inspection; defect classification; machine learning; deep learning; image processing; signal processing

check for updates

Citation: Brambilla, P.; Conese, C.; Fabris, D.M.; Chiariotti, P.; Tarabini, M. Algorithms for Vision-Based Quality Control of Circularly Symmetric Components. *Sensors* **2023**, *23*, 2539. https://doi.org/10.3390/s23052539

Academic Editor: Antonio Guerrieri

Received: 27 January 2023
Revised: 17 February 2023
Accepted: 21 February 2023
Published: 24 February 2023

1. Introduction

Nowadays, in the Industry4.0 framework, the interest in improving the quality of the products is increasing thanks to the rapid spread of new technologies and the increasing usage of machine learning (ML) and artificial intelligence (AI) approaches at the production level. The application of these new approaches to image-based quality inspection is gaining popularity, as evidenced by the increasing number of research publications in the field of quality control in the last 10 years [1].

An area of interest in several industrial fields is the identification of defects on circularly symmetric components (CSC) such as bearings, gears, buttons, circular saw blades, clutch discs, and washers [2]. These components are produced by automatic plants, and in order to achieve zero-defect manufacturing, it is not possible to adopt the classical visual inspection approach [3]. In the case of CSC, it is possible to perform specific analyses thanks to the knowledge of the symmetry and periodicity of the components.

Since the literature does not address the specific case of defect identification on knurled washers, we decided to perform a systematic literature review following the guidelines for a systematic literature review applied to engineering [4]. Queries were used to search in the academic database; each query included some keywords (“defect detection,” “defect identification,” “computer vision,” “machine vision,” “machine learning,” “deep learning”) bound by a Boolean logic operator. The research resulted in 84 unique titles. Articles were investigated to find information about the field of application, the type of hardware, the acquisition strategies, and the algorithms used.

Among these 84 documents, five are surveys and reviews [3,5–8]. In the field of application, many works can be traced back to three specific categories: 25 documents are related to the steel industry, 14 to food and agriculture, and 10 to textiles and fabric production.

Different studies aim at identifying superficial defects on non-periodic and non-symmetrical objects such as imperfection in fruits [9], crack in concrete structure [10], dent in metallic surface [7] and scratch on screen [11].

Ten works focused on the identification of defects in CSC; the majority related to the study of bearings [12–16], and some of them [12,13] exploited the transformation of the image from polar to cartesian (P2C) coordinates as proposed in this manuscript. The P2C transformation is used on other products such as the bottom side of bottles [17,18], camera lenses [19], circular polyurethane sealing elements [20], and metal cans [21].

The systematic literature review provided an overview of the data processing approaches that can be used for the generic problem of defect identification. In 18 articles [10,22–38], a deep learning (DL) approach based on neural networks (NN) was used. Some papers exploit custom DL models; many others use known architectures such as ResNet, VGG, U-Net, GoogLeNet, AlexNet, and Yolo. Other works use simple NN as multilayer perceptrons (MLP) [39,40]. For the classification task, it turned out that Support Vector Machine (SVM) is the most used algorithm; it was used in 10 works [19,26,30,33,39,41–46]. Apart from SVM, other machine learning (ML) algorithms deserve to be mentioned, e.g., k-Nearest-Neighbors (k-NN) presented in five papers [26,33,45,47,48], clustering algorithms [11,49], and random forest [45]. Out of the DL and ML approaches, many works propose a classification based on decision trees or custom algorithms [17,50–52].

Since none of the above listed works focused on washers, the research was extended to include fasteners, washers, and fixtures in the original queries. Existing studies focused on the inspection of fasteners and the detection of missing washers in assemblies [53–59] and not on the quality of washers and their defects. No dataset including labelled or non-labelled washers with defects with the desired characteristics was found: existing datasets included different washers' models, images acquired with different optical setups, often with the camera optical axis not coincident with the washer axis.

In this work, we compare the performances of standard algorithms based on feature analysis [2] with DL techniques in the specific application of quality control of knurled washers. The paper is structured as follows: The proposed DL method and the algorithms for data pre-processing are described in Section 2. The case study is presented in Section 3; Section 4 summarizes the experimental results, which are discussed in Section 5, together with the conclusion of our study.

2. Materials and Methods

2.1. Image Preprocessing

The image pre-processing consists of the identification of the specimen in the image area, the determination of the object center, and the polar to cartesian transformation. These operations are fairly standard and can be applied to any CSC according to the method described in the next few paragraphs. To demonstrate the applicability of this approach to any CSC, we will discuss it considering, as a general test case, a conical gear wheel and a ball bearing, moreover the washer, object of the case study is shown too. Given its transversal applicability, we will still refer to the component as CSC.

2.1.1. Detection of the Specimen

This step allows isolating the image pixels corresponding to the CSC from the background. The optical system must be designed so that the pixels in the background have a different grey scale level from the pixels corresponding to the CSC. In these conditions, the threshold value for the binarization can be automatically selected by the Otsu method [60], which minimizes the weighted variance between the foreground and background pixels. Examples are shown in Figure 1.

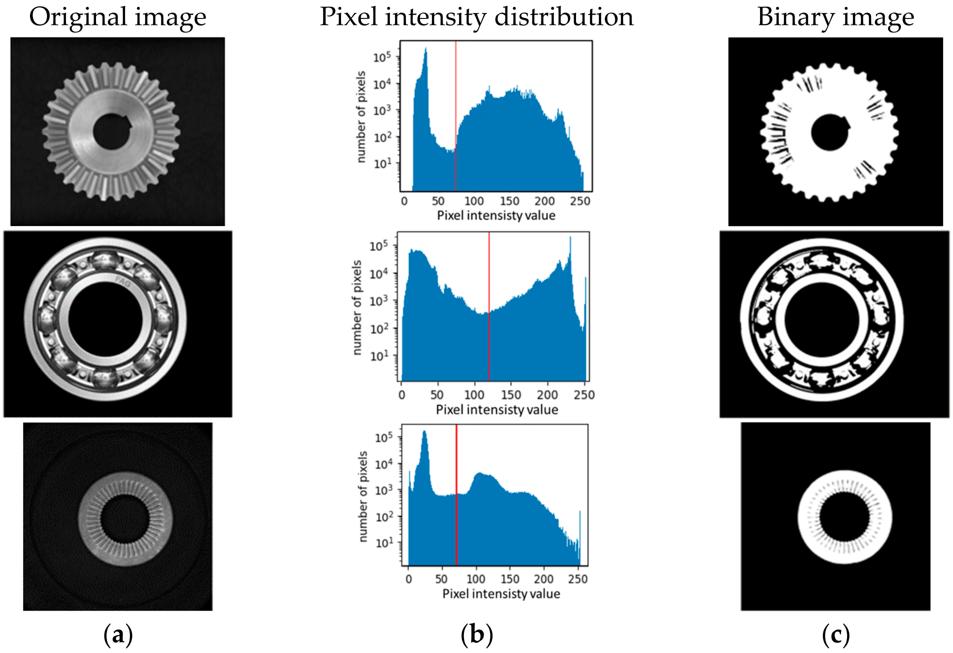


Figure 1. Automatic thresholding of a conic gear wheel (first row) and of a ball bearing (second row) and a washer (third row). Column (**a**) shows the original images; column (**b**) shows the histograms of pixel intensity distribution (the red lines are the thresholds computed with the Otsu method). Column (**c**) shows the binary images.

2.1.2. Identification of the Object Center

The second step allows identifying the center of the CSC; this passage is crucial since it determines the origin of the polar reference system. The inner and outer edges of the CSC can be detected using the Hough transform [61] for circular primitives, whose result consists of the circles' centers and diameters.

Results of the application of the Hough transform are shown in Figure 2, which illustrates the outer (green) and inner (red) circumferences and their centers. The center of the CSC (x_C, y_C), reported in blue in Figure 2, is computed as the middle point between the centers of the inner and outer circumferences; this point becomes the origin of the polar reference system.

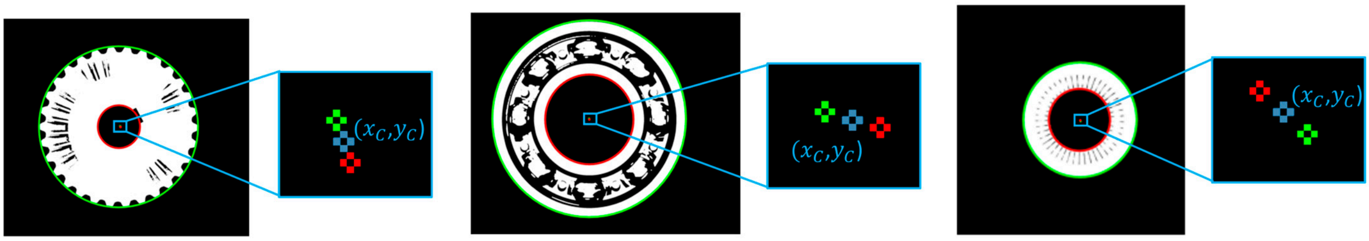


Figure 2. Circumferences found by the Hough transformation on the inner (red) and outer (green) perimeters of the binary picture of the gear wheel, ball bearing and washer. In the right part of each picture the difference in the position of the two centers is highlighted.

In this way, it is also possible to evaluate the eccentricity of the sample, which in many circularly symmetric components is considered a defect.

2.1.3. Polar to Cartesian Coordinate Transformation

As evidenced by the literature review, the CSC analysis must be performed by analyzing annuli that are transformed into stripes by the P2C transform (the CSC image is transformed into a polar coordinate system with the origin in $(x_C,\ y_C)$).

First, a square shaped Region of Interest (ROI) is extracted from the original image; the ROI's height and width are equal to the diameter of the outer circle, while the center of the square corresponds to the center of the washer. The P2C transformation equation is therefore:

$$\begin{cases} x = C_x + r \cdot cos(\theta) \\ y = C_y + r \cdot sin(\theta) \end{cases} \tag{1}$$

where:

- x is the horizontal coordinate of the original cartesian reference system;
- y is the vertical coordinate of the original cartesian reference system;
- r is the radius of the polar reference system;
- θ is the angle of the polar reference system;

The vertical coordinate of the new cartesian reference system is given by the radius itself, while the horizontal coordinate of the new reference system w is computed through Equation (2).

$$\theta = 2\pi \left(\frac{w}{W} \right) \tag{2}$$

where W is the length of the perimeter of the outer circumference. The result of the P2C transformation is shown in Figure 3. In the picture, the procedure is applied to the gear wheel and to the washer.

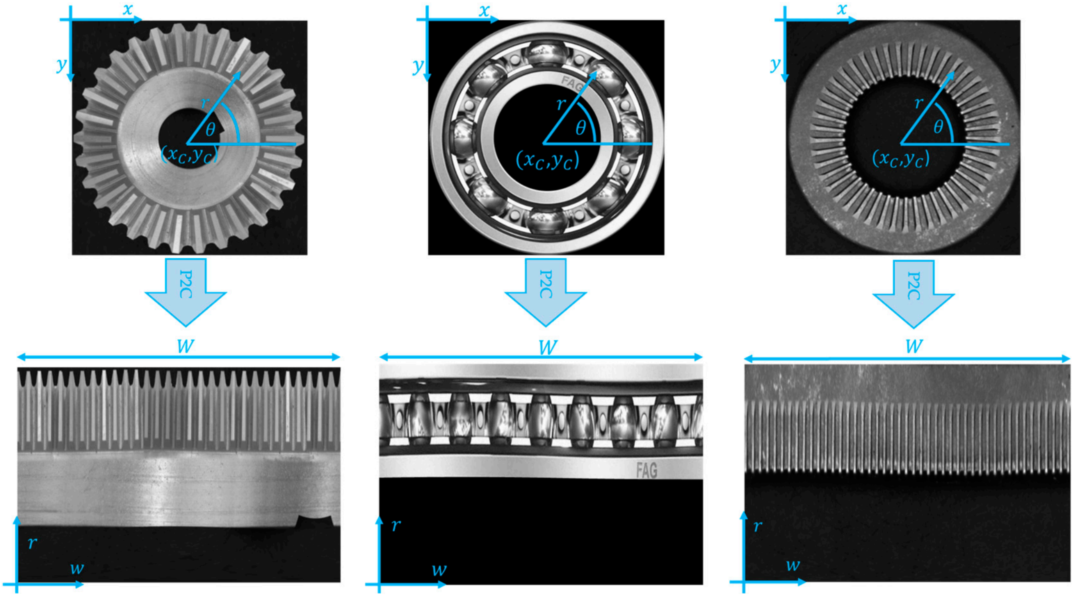


Figure 3. Polar to Cartesian coordinate transformation. The original figures in the Cartesian reference system (x, y) are shown on the upper row. Their correspondent images in the (r, w) polar coordinate system are shown on the second row. The transformation is applied on a conical gear, a ball bearing and a washer.

2.2. Standard Approach

2.2.1. Features Extraction

The image obtained after the P2C transformation is periodic, and it is possible to extract one or more pseudo-signals from the images in the r-w coordinate system by analyzing groups of rows. Said $\lambda(r,w)$ the grayscale level of the pixel with coordinates (r,w), pseudo signals (γ) can be computed as a function of the abscissa w, as the average grayscale levels between belonging to a certain stripe (hereinafter, stripe indicates group of pixels between the rows r_0 and r_{M-1}):

$$\gamma(w) = \frac{1}{M} \sum_{r=r_0}^{r_{M-1}} \lambda(r, w) \qquad w = [0, W] \tag{3}$$

where:

- γ is the pseudo-signal describing the grey-scale level along a certain group of rows
- $\gamma(w)$ is the value of the wth sample of the pseudo signal, corresponding to a specific horizontal position/angle;
- M is the height of the stripe expressed in pixels;
- r_0 is the lower coordinate of the stripe;
- r_{M-1} is the upper coordinate of the stripe.

An example of pseudo-signal extracted from the conical gear of Figure 3 is shown in Figure 4.

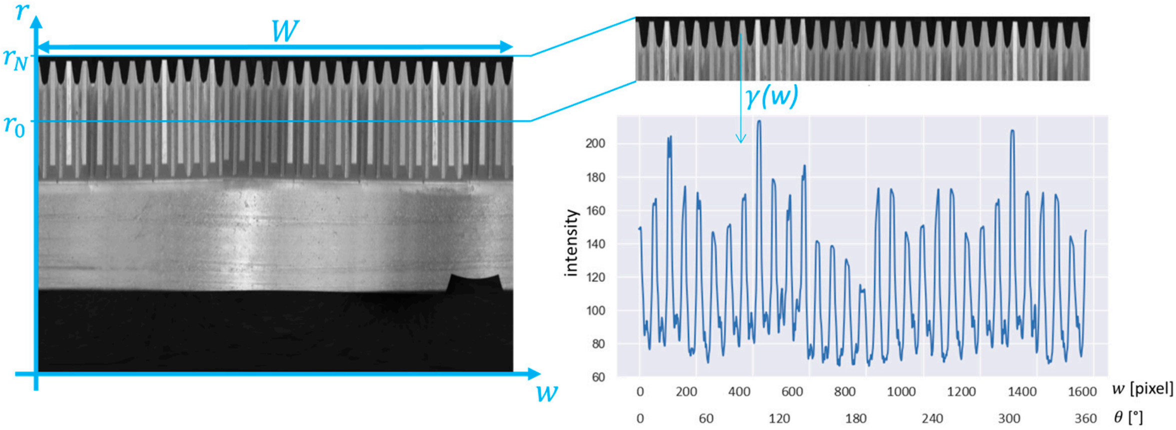


Figure 4. Procedure for the extraction of the stripe from the Cartesian picture and generation of the pseudo signal from the stripe in the case of a gear wheel with 30 teeth.

The position and the height of the stripes must be selected depending on the average size of the defect and on its expected location. Good results are usually obtained imposing M between two times and five times the size of the defect expressed in pixels.

The common features that can be extracted from $\gamma(w)$ are the root mean square (RMS), the standard deviation (SD), the skewness and kurtosis indexes, or other parameters such as the percentiles or the crest factor. An exhaustive list of the possible features that can be extracted from $\gamma(w)$ is presented in [62,63]

Since $\gamma(w)$ is a periodic signal, it can also be analyzed in the angle domain using the Fourier transform. The signal Γ can be obtained as a function of the number of repetitions per element (k) as follows:

$$\Gamma(k) = \frac{1}{W} \sum_{i=0}^{W} \gamma(i) e^{-j2\pi k \frac{i}{W}} \tag{4}$$

The resulting spectrum is shown in Figure 5. In general, the harmonic components that are multiples of the number of periodic elements are clearly visible.

The features that can be extracted from the spectrum are the amplitude of specific harmonic components (for instance, the one indicating the number of periodic components or its multiples) or the spectral centroid [62], the Mean and the Median Angular Frequency [64].

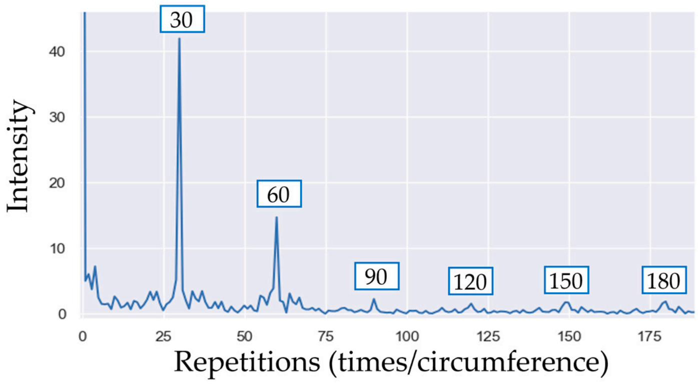


Figure 5. Fourier transform of the pseudo signal shown in Figure 4. The main peak indicates the number of teeth of the CSC (30 in the specific case of the gear wheel).

2.2.2. Feature-Based Classification

The input of the classifier is the list of features computed in the previous section, while the output is a discrete index or continuous index expressing the possibility of the presence of defects. As evidenced in the literature review, many algorithms can be used to classify the presence of defects starting from the features extracted from the pseudo signal. The most common ones are probably the SVM, NN, RF, k-NN, and clustering. The choice of the classifier depends on different aspects, such as the complexity and dimension of the input data, the dataset dimension, and the complexity of the classification task.

Many of the algorithms cited above can be tuned with different parameters (the type of kernel in the SVM, the number of neighbors in the k-NN, the number and type of nodes and layers in a NN classifier . . .). The selection of these parameters must be carried out by maximizing the classification accuracy while minimizing the computational time. In the industrial implementation, the complexity of the algorithms must be limited to allow their execution in the amount of time available for the quality inspection; within the algorithms that can respect this time constraint, the parameters that guarantee higher accuracy are selected.

2.3. Deep Learning Approach

The deep learning approach is based on the analysis of a single element of the CSC. The image obtained after the P2C transformation or the stripe (Figures 3 and 4) can be divided into N components in order to obtain a representation of a single periodic element, where N is the number of repeated elements in the CSC (N is 30 in the example of the gear wheel). As an example, Figure 6 shows the division of the image in Figure 3 into 30 parts to obtain 30 images of the periodic element.

The images obtained with this procedure are classified as “compliant” or “defective” using the DL approach. The pictures of the single element (the tooth of the gears) are input tensors of NN models trained for image classification. The classification can be performed using the state-of-the-art architectures; common choices for image classification are MobileNetV2 [65] and ResNet50 [66]. The effectiveness of the DL method in a real industrial application depends on the hyperparameter optimization, which must be chosen as a trade-off between computational time and accuracy. A sensitivity analysis versus the number of epochs, learning rate, and batch size is needed in order to derive the best classification performances in the maximum time available for quality control.

In both cases, we suggest adding data augmentation layers before the main node and the “Dense” layer: given the circular symmetry, data can be easily augmented by adding random rotations and horizontal/vertical shifts. The amounts of rotation and translation must be chosen depending on the size of the image and on the number of periodic components N. The obtained images are then normalized, i.e., the pixel values are scaled in range from 0 to 1 by dividing every pixel by the highest intensity value, i.e., 255. A "Softmax" activation is used in the final layer to perform the classification.

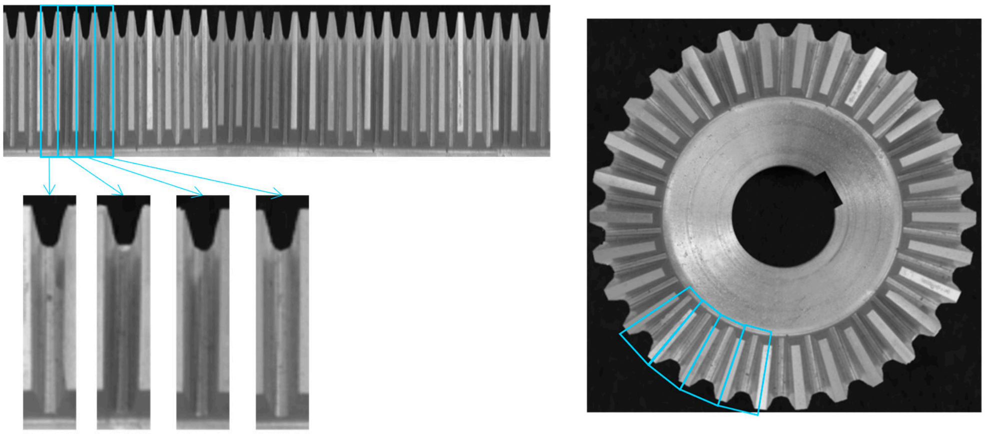


Figure 6. Tooth extraction from the stripe and visualization of the teeth in the original circularly symmetric image of the gear wheel.

3. Case Study

3.1. Standard Approach

The proposed methods were applied to monitor the quality of knurled washers produced through blanking and coining in an industrial plant. An example of the component is reported in Figure 7. The object is characterized by the presence of 45 teeth placed radially along the whole circumference of the washer. The teeth are the only area of the washer subjected to defects.

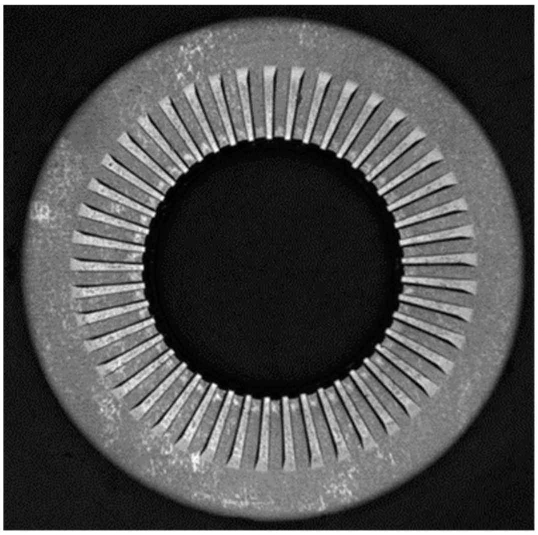


Figure 7. Knurled washer: case study component.

The pre-processing steps were applied as described in Section 2 to obtain the cartesian coordinate image.

In the standard approach, the knurled area of the washer was divided into five stripes of equal height. An example of the Cartesian image, a stripe, and the corresponding pseudo-signal are shown in Figure 8. The features extracted from $\gamma(w)$ were the root mean square value, standard deviation, skewness, and kurtosis.

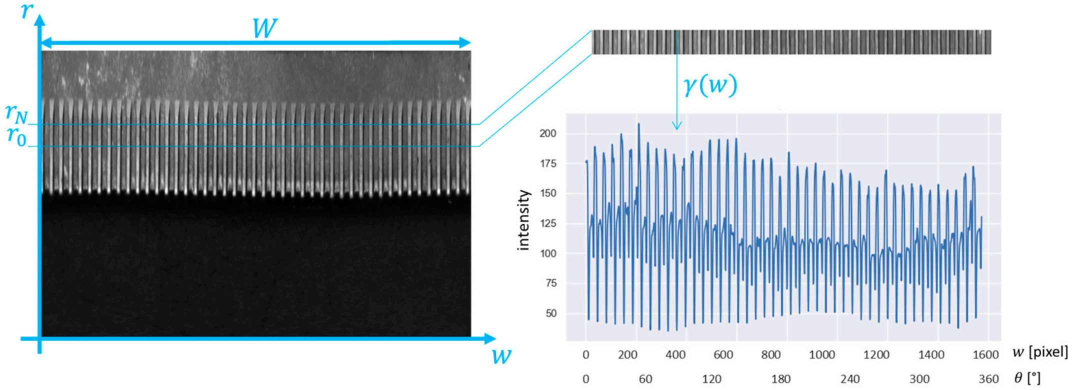


Figure 8. Procedure for the extraction of the stripe from the cartesian picture, and the construction of the pseudo signal from the stripe applied on the knurled washer.

The Fourier transform $\Gamma(k)$ of the pseudo-signal is shown in Figure 9, and the number of repeated elements in the CSC is N = 45 since the washer has 45 coined teeth. The most relevant harmonics are the multiple of 45.

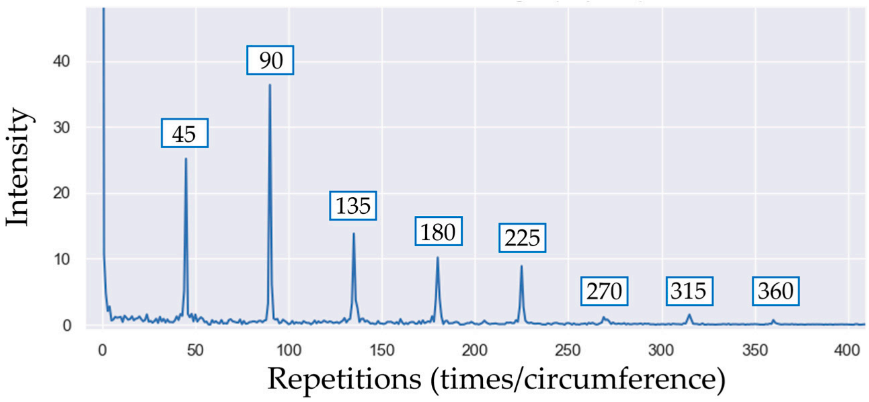


Figure 9. Angular frequency analysis of the pseudo signal shown in Figure 8.

The features extracted from $\Gamma(k)$ are the amplitudes of the first eight harmonic components and the mean angular frequency and median angular frequency as described in [64].

Since 14 features are extracted from each of the 5 stripes, each washer is represented by 70 features, which are the input variables of the binary classifier. In this specific case study, we implemented a NN and an SVM. In this implementation, the architecture of the NN is composed of 70 input nodes followed by two hidden layers with eight and four Rectifier Linear Unit (ReLU) nodes, respectively. The last layer of the network is a sigmoid node, so the value returned by the algorithm is between 0 and 1. The number of layers and nodes was manually tuned, selecting the combination granting the highest accuracy; depending on the number of features and on the complexity of the classification, other architectures can lead to better performances [67]. The SVM has been implemented the radial basis function (RBF), the kernel coefficient (gamma parameter) was left to the default value equal to $1/nf$ where nf is the number of features.

3.2. Deep Learning Approach

The teeth of the knurled washer are extracted from the cartesian picture by dividing the cartesian image into 45 parts, as shown in Figure 10.

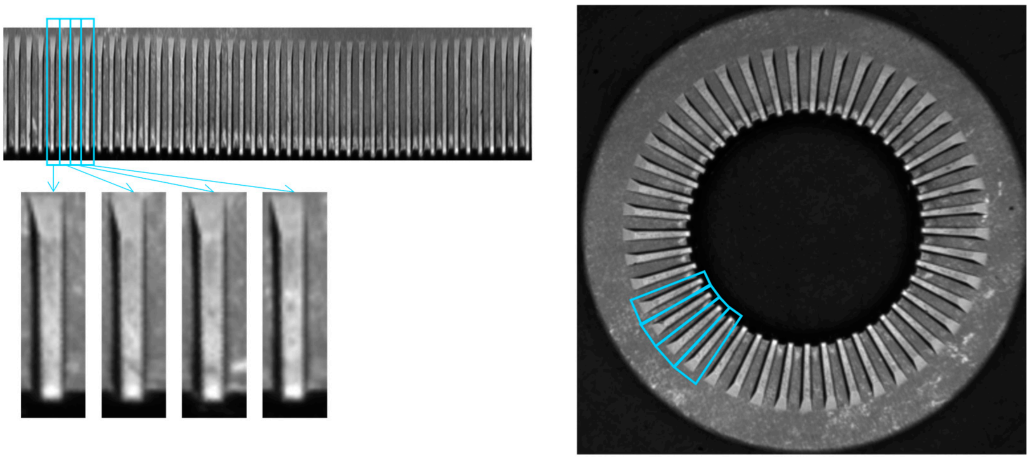


Figure 10. Teeth extraction from the stripe and visualization of the teeth on the original circularly symmetric image of the knurled washer.

An example of compliant and defective teeth is shown in Figure 11. The defective samples are sorted from the one with the biggest defect to the one with the smallest defect.

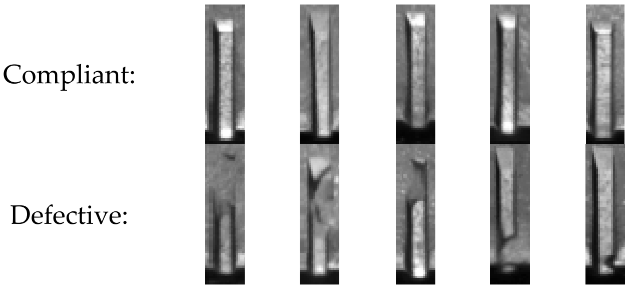


Figure 11. Example of compliant and defective teeth. The defects are sorted from the biggest to the smallest.

Images were classified using MobileNetV2 and a ResNet50, both completed with a final layer with two "Dense" nodes and "Softmax" activation so that each node represents the probability of the input sample belonging to that class. The data augmentation layer included a random rotation in the range between −5 and +5° and a random horizontal and vertical shift in the interval between −5 and +5 pixels.

The training hyperparameters, such as learning rate, number of epochs, and batch size, were selected after evaluating the classification performances versus different combinations of these parameters. The values selected in the final implementation are shown in Table 1.

Table 1. Deep Learning models hyperparameters.

Model	MobileNetV2	ResNet50
Number of epochs	125	400
Learning rate	0.0001	0.00005
Batch size	32	64

3.3. Performance Evaluation Procedure

The standard and the deep learning approaches were trained and tested on a dataset (hereinafter the washer's dataset) composed of 300 washers. In this dataset, 40% of the samples were defective, and the remaining 60% were compliant.

For the standard approach, the dataset was randomly divided into three subsets, each containing 150 pictures for training, 50 for validation, and 100 for testing.

For the deep learning approach, 200 images were split into 45 parts, including 1 tooth each, resulting in 9000 tooth pictures. In most of the cases, the defective washer contains only one or a few defective teeth; the derived dataset is unbalanced and contains 242 defective samples—corresponding to 2.7% of the population. To obtain a balanced dataset, 363 tooth samples were randomly picked from the compliant teeth sample. In this way, a dataset containing 40% defective teeth and 60% compliant teeth was obtained (Figure 12). Further, in this case, algorithm performances were tested on the remaining 100 washers.

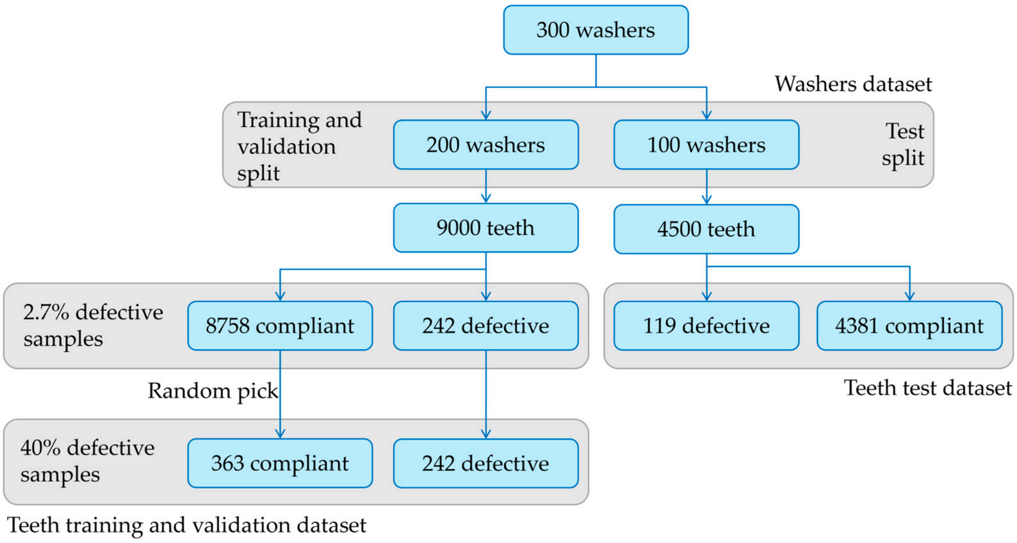


Figure 12. Teeth dataset creation flowchart.

The metrics used for the performance validation are reported in Equations (5)–(7):

$$accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{5}$$

$$Precision = \frac{TP}{(TP + FP)} \tag{6}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{7}$$

where:

- *TP* is the number of true positive classified samples;
- *TN* is the number of true negative classified samples;
- *FP* is the number of false positives classified as samples;
- *FN* is the number of false negative classified samples.

The ground truth was determined by an expert operator of the manufacturing company who actually works as a quality inspector.

4. Results

The accuracy, precision, and recall obtained with the standard approach are shown in Table 2; and confusion matrices are reported in Appendix A (Figure A1). The results of the SVM and feature-based NN classifiers are compared.

Table 2. Classification performances: standard algorithm.

Model	SVM	NN
Accuracy	0.97	0.98
Precision	0.95	0.98
Recall	0.97	0.98

Table 3 shows the accuracy, precision, and recall obtained by the DL models over the test split of the teeth dataset. The confusion matrices are reported in Appendix A; performances represent the performance of the network in the detection of dental defects.

Table 3. Classification performances: Deep learning approaches.

Model	MobileNetV2	ResNet50
Accuracy	0.997	0.997
Precision	0.928	0.907
Recall	0.975	0.992

The comparison of performances between the standard algorithms and the DL ones is not straightforward because of the different datasets ("washer" vs. "teeth"); nevertheless, a washer is defective if at least one tooth is defective. Since the misclassification of a single tooth in most of the cases means the misclassification of the entire washer (it does not happen if two teeth misclassified belong to the same washer) the performances of the DL algorithms are manually recomputed based on a "washer-based" classification. Table 4 shows the results.

Table 4. Performances of the DL algorithms computed based on the washer classification.

Model	MobileNetV2	ResNet50
Accuracy	0.890	0.890
Precision	0.796	0.822
Recall	0.975	0.925

The chart in Figure 13 shows the training history of the MobileNetV2 model; on the left, training (blue) and validation (orange) accuracy variation over the training epochs is reported, and on the right, training (blue) and validation (orange) loss evolution is displayed.

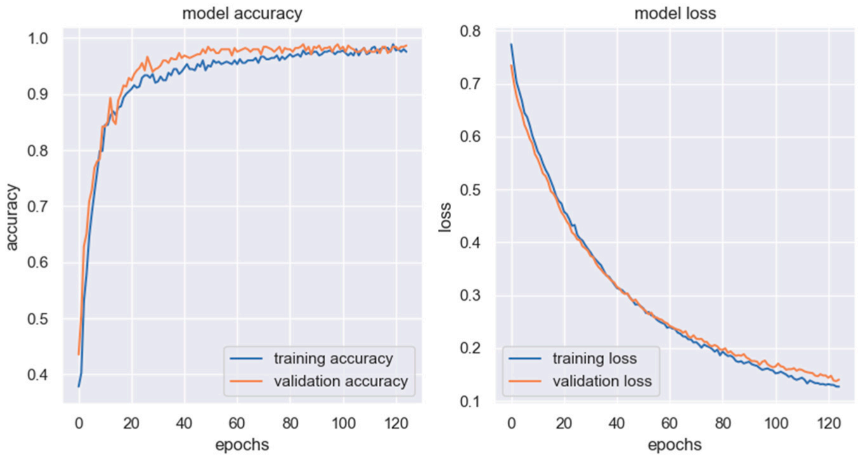


Figure 13. Training history of MobileNetV2 architecture.

Figure 14 shows the evolution of the same variables for the ResNet50 model: on the left, the training (blue) and validation (accuracy); on the right, the training (blue) and validation (orange) loss.

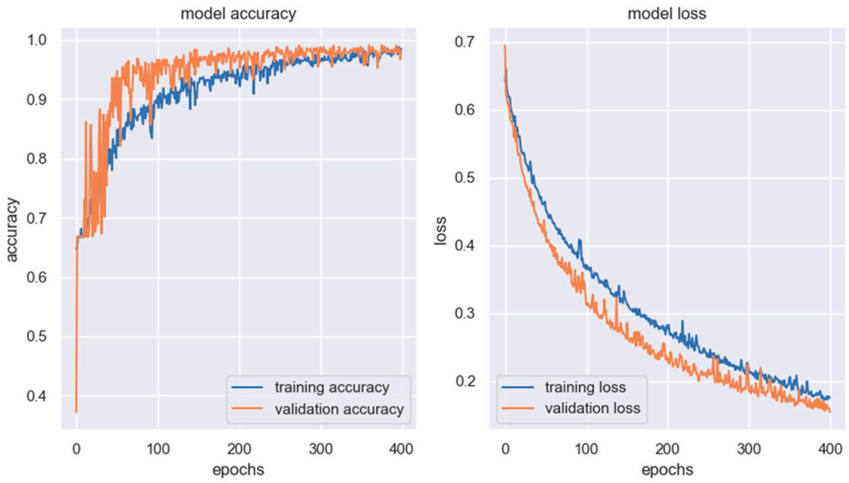
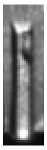
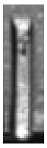
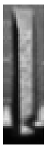
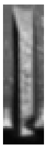


Figure 14. Training history of ResNet50 architecture.

Table 5 shows the result of the classification for some critical cases where the two models classified the specific tooth in different ways:

Table 5. Misclassified teeth by different models.

	Actual Class	Picture	MoblieNetV2	ResNet50
(a)	Compliant		Compliant	Defective
(b)	Compliant		Compliant	Defective
(c)	Defective		Compliant	Defective
(d)	Defective		Compliant	Defective

The computational effort for all the approaches was also evaluated using a personal PC with an Intel Core i7 processor and 16 GB of RAM. Analyses were performed to identify the time requested for pre-processing the image, for the pseudo-signal and tooth extraction, and for the classification of the image (defective/non defective).

- The time for the pre-processing routine common for the two approaches is 191.9 ± 9.8 ms (C.I. 95%), computed over the 200 samples of Washers' training and validation split.
- The time for the computation of the pseudo signal is lower than 2 ms; a similar value is required to extract the teeth for the DL approaches.
- The time for the feature-based classification of a washer using the standard algorithm is lower than 1 ms for both the SVM classifier and the NN-based approach.
- The time for the classification of a tooth is 13.28 ± 0.06 ms (C.I. 95%) for the MobileNetV2 model and 36.71 ± 0.08 ms (C.I. 95%) for the ResNet50, both values were computed on 4500 samples of the teeth test dataset.

The overall time requested for the classification of an image using the standard approach is, on average, 1937 ms and is dominated by the image pre-processing. The average time requested for the classification using DL approaches is between 2075 ms (MobileNetV2) and 2306 ms (ResNet50) in the case of parallelization of the teeth classification and between 789 ms (MobileNetV2) and 1843 ms (ResNet50) in the case of serial classification of the 45 teeth.

Another aspect related to the computational effort is related to the algorithm training. Table 6 reports the number of training epochs for each algorithm, the time required to train one epoch, and the total training time. The models training gets advantages from GPU acceleration.

Table 6. Training time of the different algorithms.

Algorithm	N° of Epochs	Training Time (One Epoch)	Total Training Time
Standard algorithm—SVM	–	–	<1 s
Standard algorithm—NN	20	<1 s	2 s
DL—MobileNetV2	125	13 s	27 min
DL—ResNet50	400	33 s	220 min

5. Discussion and Conclusions

This paper proposes different methods for the binary classification (damaged/undamaged) of CSC. In the first approach, a custom algorithm is used to extract features from pseudo-signals representing the pixel-wise intensity of image subsets. The second approach consists in the extraction of a ROIs from the image of the specimen; each ROI contains a potentially defective element of the component.

The proposed methods are tested on a case study of a knurled conical washer containing 45 teeth distributed along its circumference; both approaches correctly identify the defective components among the compliant ones.

The performances of the DL approaches on the teeth test dataset evidence the high accuracy and the lower precision and recall. The behavior is due to the strongly unbalanced dataset and the consequent presence of a high number of correctly predicted compliant samples (class 0). The results show that ResNet50 is more sensitive to defects but leads to a higher number of false positives with respect to MobileNetV2. False positive predictions occurred when samples were characterized by the presence of dust that was wrongly identified as a defect. The MobileNetV2 architecture is less sensitive to defects and leads to fewer false positives and more false negatives with respect to the ResNet50.

From the qualitative analysis of the training history of the two DL models, one may assume that the ResNest50 architecture needs a higher number of training epochs to get to a tuned model, while the MobileNetV2 is faster; moreover, the evolution of the accuracy and the loss of the ResNet50 model is subjected to higher variations in consecutive steps. This is a consequence of the fact that the tuning of the training parameters (i.e., learning rate, number of epochs, batch dimension, type of loss function) was carried out manually.

Considering a possible online implementation of the system, the execution time plays a key role in the identification of the optimal strategy. The time required for the classification by the standard algorithm is negligible with respect to the time needed for the image pre-processing (that can be optimized, for instance, by ensuring constant positioning of the system or optimizing the image resolution). On the contrary, as the DL approaches, the time needed for the prediction is not negligible. Even if it is low, it must be considered that it should be executed 45 times for each washer, whereas the standard algorithm requires just one single execution for each washer. This issue can be easily solved by parallelizing the classification on different cores since the classification of a tooth is completely independent from the classification of any other tooth.

The greatest advantage of the DL approach lies in the scalability of the problem. Once the NN is trained, it can be used to classify the specific element independently based on the number of occurrences in the component. For example, once the algorithms are trained for the classification of teeth on the washer, the same models can be used for another washer model with a different number of teeth, given the tooth shape is the same.

Author Contributions: Conceptualization, M.T.; methodology, P.B., C.C., P.C. and M.T.; software, P.B., C.C. and D.M.F.; investigation, P.B.; data curation, P.B.; visualization, P.B.; writing—original draft preparation, P.B. and C.C.; writing—review and editing, P.C.; supervision, P.C. and M.T. All authors have read and agreed to the published version of the manuscript.

Funding: The activity was funded by Growermetal SPA.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy reasons.

Acknowledgments: The authors acknowledge Growermetal S.p.A. for funding the project within the activities of the Joint Research Center MATT.

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

Abbreviations

Here the list of abbreviations used in this manuscript.

Abbreviation	Definition
AI	Artificial Intelligence
CSC	Circularly Symmetric Component
DL	Deep Learning
K	Kurtosis
k-NN	k-Nearest-Neighbors
MAF	Mean Angular Frequency
MDAF	Median Angular Frequency
ML	Machine Learning
MLP	Multilayer Perceptron
NN	Neural Network
P2C	Polar to Cartesian
RMS	Root Mean Square
ROI	Region of Interest
SD	Standard Deviation
SK	Skewness
SVM	Support Vector Machine

Appendix A

In this section, the confusion matrices of the classification algorithms are reported.

Figure A1 shows the confusion matrices of the SVM classifier (a) and the NN classifier (b) with the standard algorithm evaluated on the test split of the “Washer’s dataset”.

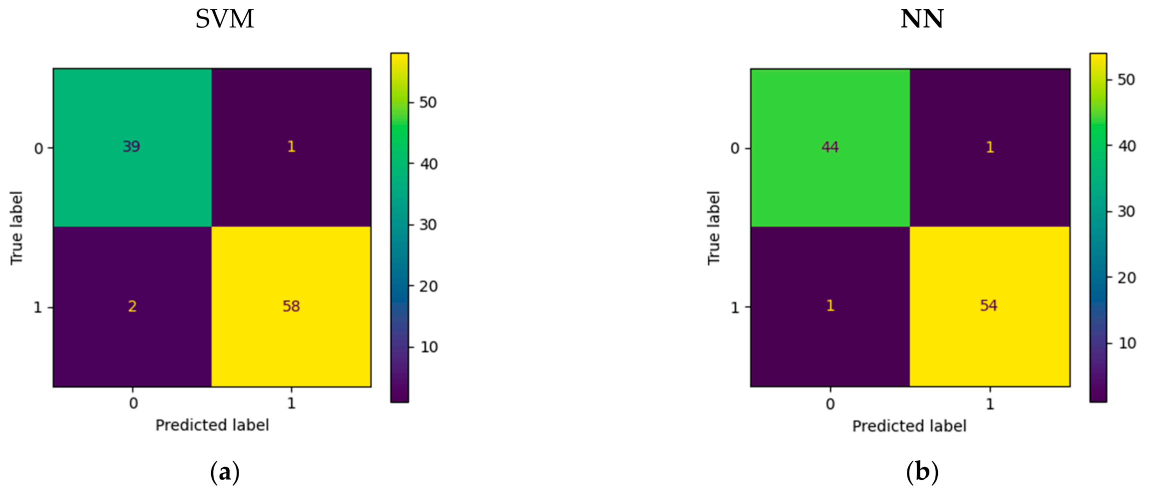


Figure A1. Standard algorithm confusion matrixes; (**a**) SVM classifier; (**b**) NN classifier.

Figure A2 shows the confusion matrices obtained with the DL approaches. Model based on MobileNetV2 (a) and ResNet50 (b) is compared. Dates are obtained on the test split of the "Teeth dataset".

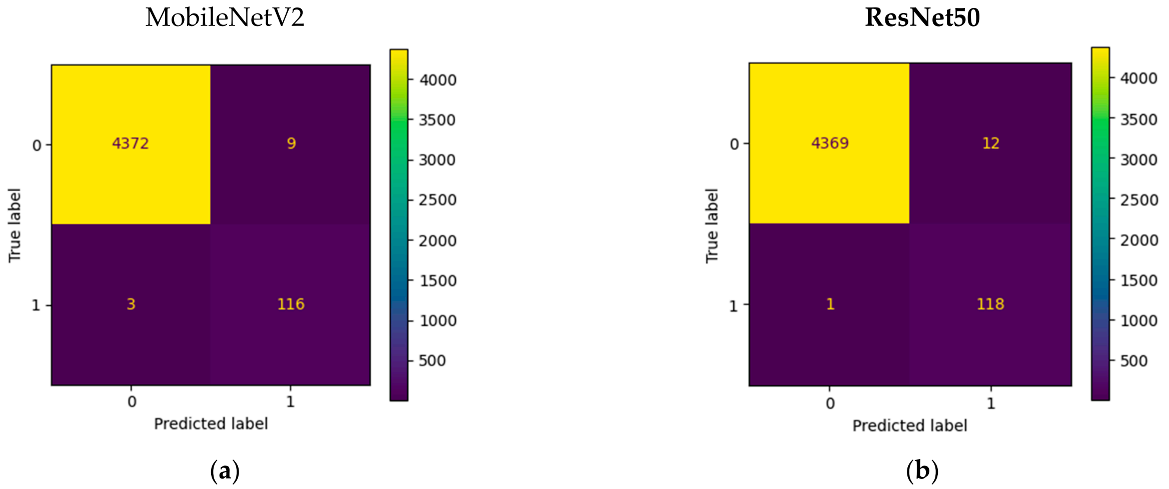


Figure A2. Deep Learning approach confusion matrixes; (**a**) MobileNetV2 model; (**b**) ResNet50 model.

References

1. Babic, M.; Farahani, M.A.; Wuest, T. Image based quality inspection in smart manufacturing systems: A literature review. *Procedia CIRP* **2021**, *103*, 262–267. [CrossRef]
2. Brambilla, P.; Cattaneo, P.; Fumagalli, A.; Chiariotti, P.; Tarabini, M. Automated Vision Inspection of Critical Steel Components based on Signal Analysis Extracted form Images. In Proceedings of the 2022 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0&IoT), Trento, Italy, 7–9 June 2022; pp. 1–5. [CrossRef]
3. Huang, S.H.; Pan, Y.C. Automated visual inspection in the semiconductor industry: A survey. *Comput. Ind.* **2015**, *66*, 1–10. [CrossRef]
4. Torres-Carrión, P.V.; González-González, C.S.; Aciar, S.; Rodríguez-Morales, G. Methodology for systematic literature review applied to engineering and education. In Proceedings of the 2018 IEEE Global engineering education conference (EDUCON), Santa Cruz de Tenerife, Spain, 17–20 April 2018; pp. 1364–1373. [CrossRef]
5. Ren, Z.; Fang, F.; Yan, N.; Wu, Y. State of the art in defect detection based on machine vision. *Int. J. Precis. Eng. Manuf.-Green Technol.* **2022**, *9*, 661–691. [CrossRef]
6. Rasheed, A.; Zafar, B.; Rasheed, A.; Ali, N.; Sajid, M.; Dar, S.H.; Habib, U.; Shehryar, T.; Mahmood, M.T. Fabric Defect Detection Using Computer Vision Techniques: A Comprehensive Review. *Math. Probl. Eng.* **2020**, *2020*, 8189403. [CrossRef]
7. Tang, B.; Chen, L.; Sun, W.; Lin, Z.K. Review of surface defect detection of steel products based on machine vision. *IET Image Process.* **2023**, *17*, 303–322. [CrossRef]

8. Koch, C.; Georgieva, K.; Kasireddy, V.; Akinci, B.; Fieguth, P. A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. *Adv. Eng. Inform.* **2015**, *29*, 196–210. [CrossRef]
9. Miller, B.K.; Delwiche, M.J. Peach defect detection with machine vision. *Trans. ASAE* **1991**, *34*, 2588–2597. [CrossRef]
10. Li, D.; Xie, Q.; Gong, X.; Yu, Z.; Xu, J.; Sun, Y.; Wang, J. Automatic defect detection of metro tunnel surfaces using a vision-based inspection system. *Adv. Eng. Inform.* **2021**, *47*, 101206. [CrossRef]
11. Jian, C.; Gao, J.; Ao, Y. Automatic surface defect detection for mobile phone screen glass based on machine vision. *Appl. Soft Comput. J.* **2017**, *52*, 348–358. [CrossRef]
12. Deng, S.; Cai, W.; Xu, Q.; Liang, B. Defect detection of bearing surfaces based on machine vision technique. In Proceedings of the 2010 International Conference on Computer Application and System Modeling, ICCASM 2010, Taiyuan, China, 22–24 October 2010; Volume 4. [CrossRef]
13. Shen, H.; Li, S.; Gu, D.; Chang, H. Bearing defect inspection based on machine vision. *Measurement* **2012**, *45*, 719–733. [CrossRef]
14. Liao, D.; Yin, M.; Luo, H.; Li, J.; Wu, N. Machine vision system based on a coupled image segmentation algorithm for surface-defect detection of a Si 3 N 4 bearing roller. *JOSA A* **2022**, *39*, 571. [CrossRef] [PubMed]
15. Wang, J.; Qiao, J.; Guo, M. Research on bearing surface defect detection system based on machine vision. *J. Phys. Conf. Ser.* **2022**, *2290*, 012061. [CrossRef]
16. Gu, Z.; Liu, X.; Wei, L. A Detection and Identification Method Based on Machine Vision for Bearing Surface Defects. In Proceedings of the ICCCR 2021—2021 International Conference on Computer Control and Robotics, Shanghai, China, 8–10 January 2021; pp. 128–132. [CrossRef]
17. Zhou, X.; Wang, Y.; Xiao, C.; Zhu, Q.; Lu, X.; Zhang, H.; Ge, J.; Zhao, H. Automated Visual Inspection of Glass Bottle Bottom with Saliency Detection and Template Matching. *IEEE Trans. Instrum. Meas.* **2019**, *68*, 4253–4267. [CrossRef]
18. Zhou, X.; Wang, Y.; Zhu, Q.; Mao, J.; Xiao, C.; Lu, X.; Zhang, H. A Surface Defect Detection Framework for Glass Bottle Bottom Using Visual Attention Model and Wavelet Transform. *IEEE Trans. Industr. Inform.* **2020**, *16*, 2189–2201. [CrossRef]
19. Chang, C.-F.; Wu, J.-L.; Chen, K.-J.; Hsu, M.-C. A hybrid defect detection method for compact camera lens. *Adv. Mech. Eng.* **2017**, *9*, 168781401772294. [CrossRef]
20. Chiou, Y.-C.; Li, W.-C. Flaw detection of cylindrical surfaces in PU-packing by using machine vision technique. *Measurement* **2009**, *42*, 989–1000. [CrossRef]
21. Chen, T.; Wang, Y.; Xiao, C.; Wu, Q.M.J. A machine vision apparatus and method for can-end inspection. *IEEE Trans. Instrum. Meas.* **2016**, *65*, 2055–2066. [CrossRef]
22. Paraskevoudis, K.; Karayannis, P.; Koumoulos, E.P. Real-time 3d printing remote defect detection (Stringing) with computer vision and artificial intelligence. *Processes* **2020**, *8*, 1464. [CrossRef]
23. Wang, J.; Fu, P.; Gao, R.X. Machine vision intelligence for product defect inspection based on deep learning and Hough transform. *J. Manuf. Syst.* **2019**, *51*, 52–60. [CrossRef]
24. Li, R.; Yuan, Y.; Zhang, W.; Yuan, Y. Unified Vision-Based Methodology for Simultaneous Concrete Defect Detection and Geolocalization. *Comput.-Aided Civ. Infrastruct. Eng.* **2018**, *33*, 527–544. [CrossRef]
25. Fuchs, P.; Kröger, T.; Garbe, C.S. Defect detection in CT scans of cast aluminum parts: A machine vision perspective. *Neurocomputing* **2021**, *453*, 85–96. [CrossRef]
26. Mery, D.; Arteta, C. Automatic defect recognition in x-ray testing using computer vision. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017, Santa Rosa, CA, USA, 24–31 March 2017; pp. 1026–1035. [CrossRef]
27. Li, C.; Zhang, X.; Huang, Y.; Tang, C.; Fatikow, S. A novel algorithm for defect extraction and classification of mobile phone screen based on machine vision. *Comput. Ind. Eng.* **2020**, *146*, 106530. [CrossRef]
28. Deng, L.; Li, J.; Han, Z. Online defect detection and automatic grading of carrots using computer vision combined with deep learning methods. *LWT* **2021**, *149*, 111832. [CrossRef]
29. Jeyaraj, P.R.; Samuel Nadar, E.R. Computer vision for automatic detection and classification of fabric defect employing deep learning algorithm. *Int. J. Cloth. Sci. Technol.* **2019**, *31*, 510–521. [CrossRef]
30. Huang, S.; Fan, X.; Sun, L.; Shen, Y.; Suo, X. Research on Classification Method of Maize Seed Defect Based on Machine Vision. *J. Sens.* **2019**, *2019*, 2716975. [CrossRef]
31. Yuan, Z.C.; Zhang, Z.T.; Su, H.; Zhang, L.; Shen, F.; Zhang, F. Vision-Based Defect Detection for Mobile Phone Cover Glass using Deep Neural Networks. *Int. J. Precis. Eng. Manuf.* **2018**, *19*, 801–810. [CrossRef]
32. Suo, X.; Liu, J.; Dong, L.; Shengfeng, C.; Enhui, L.; Ning, C. A machine vision-based defect detection system for nuclear-fuel rod groove. *J. Intell. Manuf.* **2022**, *33*, 1649–1663. [CrossRef]
33. Ahmadi, B.; Heredia, R.; Shahbazmohamadi, S.; Shahbazi, Z. Non-destructive automatic die-level defect detection of counterfeit microelectronics using machine vision. *Microelectron. Reliab.* **2020**, *114*, 113893. [CrossRef]
34. Aslam, M.; Khan, T.M.; Naqvi, S.S.; Holmes, G.; Naffa, R. On the Application of Automated Machine Vision for Leather Defect Inspection and Grading: A Survey. *IEEE Access* **2019**, *7*, 176065–176086. [CrossRef]
35. Duan, X.; Duan, F.; Han, F. Study on surface defect vision detection system for steel plate based on virtual instrument technology. In Proceedings of the 2011 International Conference on Control, Automation and Systems Engineering, CASE 2011, Singapore, 30–31 July 2011; pp. 1–4. [CrossRef]

36. Ficzere, M.; Mészáros, L.A.; Kállai-Szabó, N.; Kovács, A.; Antal, I.; Nagy, Z.K.; Galata, D.L. Real-time coating thickness measurement and defect recognition of film coated tablets with machine vision and deep learning. *Int. J. Pharm.* **2022**, *623*, 121957. [CrossRef]
37. Chao-Ching, H.; 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]
38. Li, C.; Huang, Y.; Li, H.; Zhang, X. A weak supervision machine vision detection method based on artificial defect simulation. *Knowl. Based Syst.* **2020**, *208*, 106466. [CrossRef]
39. Moallem, P.; Razmjooy, N.; Ashourian, M. Computer vision-based potato defect detection using neural networks and support vector machine. *Int. J. Robot. Autom.* **2013**, *28*, 137–145. [CrossRef]
40. Stavropoulos, P.; Papacharalampopoulos, A.; Petridis, D. A vision-based system for real-time defect detection: A rubber compound part case study. *Procedia. CIRP* **2020**, *93*, 1230–1235. [CrossRef]
41. Jia, H.; Murphey, Y.L.; Shi, J.; Chang, T.S. An intelligent real-time vision system for surface defect detection. In Proceedings of the International Conference on Pattern Recognition, Cambridge, UK, 26–26 August 2004; volume 3; pp. 239–242. [CrossRef]
42. Zhou, Q.; Chen, R.; Huang, B.; Liu, C.; Yu, J.; Yu, X. An automatic surface defect inspection system for automobiles using machine vision methods. *Sensors* **2019**, *19*, 644. [CrossRef]
43. Ireri, D.; Belal, E.; Okinda, C.; Makange, N.; Ji, C. A computer vision system for defect discrimination and grading in tomatoes using machine learning and image processing. *Artif. Intell. Agric.* **2019**, *2*, 28–37. [CrossRef]
44. Han, Y.; Liu, Z.; Lee, D.J.; Liu, W.; Chen, J.; Han, Z. Computer vision–based automatic rod-insulator defect detection in high-speed railway catenary system. *Int. J. Adv. Robot Syst.* **2018**, *15*, 1729881418773943. [CrossRef]
45. Nguyen, V.H.; Pham, V.H.; Cui, X.; Ma, M.; Kim, H. Design and evaluation of features and classifiers for oled panel defect recognition in machine vision. *J. Inf. Telecommun.* **2017**, *1*, 334–350. [CrossRef]
46. Jawahar, M.; Babu, N.C.; Vani, K.L.J.A.; Anbarasi, L.J.; Geetha, S. Vision based inspection system for leather surface defect detection using fast convergence particle swarm optimization ensemble classifier approach. *Multimed. Tools. Appl.* **2021**, *80*, 4203–4235. [CrossRef]
47. García, M.; Candelo-Becerra, J.E.; Hoyos, F.E. Quality and defect inspection of green coffee beans using a computer vision system. *Appl. Sci.* **2019**, *9*, 4195. [CrossRef]
48. Chen, L.; Liang, Y.; Wang, K. Inspection of rail surface defect based on machine vision system. In Proceedings of the 2nd International Conference on Information Science and Engineering, ICISE2010, Hangzhou, China, 4–6 December 2010; pp. 3793–3796. [CrossRef]
49. Wang, L.; Zhao, Y.; Zhou, Y.; Hao, J. Calculation of flexible printed circuit boards (FPC) global and local defect detection based on computer vision. *Circuit World* **2016**, *42*, 49–54. [CrossRef]
50. Sun, J.; Li, C.; Wu, X.J.; Palade, V.; Fang, W. An Effective Method of Weld Defect Detection and Classification Based on Machine Vision. *IEEE Trans. Ind. Inf.* **2019**, *15*, 6322–6333. [CrossRef]
51. Wen, Z.; Tao, Y. Building a rule-based machine-vision system for defect inspection on apple sorting and packing lines. *Expert Syst. Appl.* **1999**, *16*, 307–313. [CrossRef]
52. Praveen Kumar, R.; Deivanathan, R.; Jegadeeshwaran, R. Welding defect identification with machine vision system using machine learning. *J. Phys. Conf. Ser.* **2021**, *1716*, 012023. [CrossRef]
53. Bhatt, P.M.; Malhan, R.K.; Rajendran, P.; Shah, B.C.; Thakar, S.; Yoon, Y.J.; Gupta, S.K. Image-Based Surface Defect Detection Using Deep Learning: A Review. *J. Comput. Inf. Sci. Eng.* **2021**, *21*, 040801. [CrossRef]
54. Taheritanjani, S.; Haladjian, J.; Bruegge, B. Fine-Grained Visual Categorization of Fasteners in Overhaul Processes. In Proceedings of the 2019 5th International Conference on Control, Automation and Robotics (ICCAR), Beijing, China, 19–22 April 2019; pp. 241–248. [CrossRef]
55. Volkau, I.; Mujeeb, A.; Dai, W.; Erdt, M.; Sourin, A. The Impact of a Number of Samples on Unsupervised Feature Extraction, Based on Deep Learning for Detection Defects in Printed Circuit Boards. *Future Internet* **2022**, *14*, 8. [CrossRef]
56. Gong, H.; Deng, X.; Liu, J.; Huang, J. Quantitative loosening detection of threaded fasteners using vision-based deep learning and geometric imaging theory. *Autom. Constr.* **2022**, *133*, 104009. [CrossRef]
57. Aytekin, C.; Rezaeitabar, Y.; Dogru, S.; Ulusoy, I. Railway Fastener Inspection by Real-Time Machine Vision. *IEEE Trans. Syst. Man. Cybern Syst.* **2015**, *45*, 1101–1107. [CrossRef]
58. Liu, J.; Huang, Y.; Zou, Q.; Tian, M.; Wang, S.; Zhao, X.; Dai, P.; Ren, S. Learning Visual Similarity for Inspecting Defective Railway Fasteners. *IEEE Sens. J.* **2019**, *19*, 6844–6857. [CrossRef]
59. Chen, J.; Liu, Z.; Wang, H.; Nunez, A.; Han, Z. Automatic Defect Detection of Fasteners on the Catenary Support Device Using Deep Convolutional Neural Network. *IEEE Trans. Instrum. Meas.* **2018**, *67*, 257–269. [CrossRef]
60. Otsu, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. Syst. Man. Cybern* **1979**, *9*, 62–66. [CrossRef]
61. Duda, R.O.; Hart, P.E. Use of the Hough transformation to detect lines and curves in pic-tures. *Commun. ACM* **1972**, *15*, 11–15. [CrossRef]
62. Altın, C.; Er, O. Comparison of Different Time and Frequency Domain Feature Extraction Methods on Elbow Gesture's EMG. *Eur. J. Interdiscip. Stud.* **2016**, *2*, 35–44. [CrossRef]
63. Bendat, J.S.; Piersol, A.G. *Random Data: Analysis and Measurement Procedures*; Institutions and Businesses: Singapore, 2011.

64. Phinyomark, A.; Thongpanja, S.; Hu, H.; Phukpattaranont, P.; Limsakul, C. The Usefulness of Mean and Median Frequencies in Electromyography Analysis. In *Computational Intelligence in Electromyography Analysis—A Perspective on Current Applications and Future Challenges*; In Tech: Rijeka, Croatia, 2012. [CrossRef]
65. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. January 2018. Available online: https://arxiv.org/abs/1801.04381 (accessed on 10 December 2022).
66. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. December 2015. Available online: https://arxiv.org/abs/1512.03385 (accessed on 10 December 2022).
67. Stathakis, D. How many hidden layers and nodes? *Int. J. Remote Sens.* **2009**, *30*, 2133–2147. [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.