

Machine Learning and image analysis towards improved energy management in Industry 4.0: a practical case study on quality control

*Original*

Machine Learning and image analysis towards improved energy management in Industry 4.0: a practical case study on quality control / Casini, Mattia; De Angelis, Paolo; Porrati, Marco; Vigo, Paolo; Fasano, Matteo; Chiavazzo, Eliodoro; Bergamasco, Luca. - In: ENERGY EFFICIENCY. - ISSN 1570-646X. - ELETTRONICO. - 17:5(2024), pp. 1-16.  
[10.1007/s12053-024-10228-7]

*Availability:*

This version is available at: 11583/2988868 since: 2024-05-21T11:03:07Z

*Publisher:*

Springer Nature

*Published*

DOI:10.1007/s12053-024-10228-7

*Terms of use:*


This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)



# Machine Learning and image analysis towards improved energy management in Industry 4.0: a practical case study on quality control

Mattia Casini · Paolo De Angelis ·  
Marco Porrati · Paolo Vigo · Matteo Fasano ·  
Eliodoro Chiavazzo · Luca Bergamasco 

Received: 22 July 2023 / Accepted: 28 April 2024  
© The Author(s) 2024

**Abstract** With the advent of Industry 4.0, Artificial Intelligence (AI) has created a favorable environment for the digitalization of manufacturing and processing, helping industries to automate and optimize operations. In this work, we focus on a practical case study of a brake caliper quality control operation, which is usually accomplished by human inspection and requires a dedicated handling system, with a slow production rate and thus inefficient energy usage. We report on a developed Machine Learning (ML) methodology, based on Deep Convolutional Neural Networks (D-CNNs), to automatically extract information from images, to automate the process. A complete workflow has been developed on the target industrial test case. In order to find the best compromise between accuracy and computational demand of the model, several D-CNNs architectures have been tested. The results show that, a judicious choice of the ML model with a proper training, allows a fast and accurate quality control; thus, the proposed workflow could be implemented for an

ML-powered version of the considered problem. This would eventually enable a better management of the available resources, in terms of time consumption and energy usage.

**Keywords** Industry 4.0 · Energy management · Artificial intelligence · Machine learning · Deep learning · Convolutional neural networks · Computer vision

## Introduction

An efficient use of energy resources in industry is key for a sustainable future (Bilgen, 2014; Ocampo-Martinez et al., 2019). The advent of Industry 4.0, and of Artificial Intelligence, have created a favorable context for the digitalisation of manufacturing processes. In this view, Machine Learning (ML) techniques have the potential for assisting industries in a better and smart usage of the available data, helping to automate and improve operations (Mazzei & Ramjattan, 2022; Narciso & Martins, 2020). For example, ML tools can be used to analyze sensor data from industrial equipment for predictive maintenance (Carvalho et al., 2019; Dalzochio et al., 2020), which allows identification of potential failures in advance, and thus to a better planning of maintenance operations with reduced downtime. Similarly, energy consumption optimization (Qin et al., 2020; Shen et al., 2020) can be achieved via ML-

---

**Supplementary Information** The online version contains supplementary material available at <https://doi.org/10.1007/s12053-024-10228-7>.

---

L. Bergamasco (✉) · M. Casini · P. De Angelis · P. Vigo ·  
M. Fasano · E. Chiavazzo  
Department of Energy, Politecnico di Torino, Turin, Italy  
luca.bergamasco@polito.it

M. Porrati  
R&D Department, GEFIT S.p.a., Alessandria, Italy

enabled analysis of available consumption data, with consequent adjustments of the operating parameters, schedules, or configurations to minimize energy consumption while maintaining an optimal production efficiency. Energy consumption forecast (Liu et al., 2019; Zhang et al., 2018) can also be improved, especially in industrial plants relying on renewable energy sources (Bologna et al., 2020; Ismail et al., 2021), by analysis of historical data on weather patterns and forecast, to optimize the usage of energy resources, avoid energy peaks, and leverage alternative energy sources or storage systems (Fasano et al., 2019; Li & Zheng, 2016; Mishra et al., 2023; Ribezzo et al., 2022; Trezza et al., 2022). Finally, ML tools can also serve for fault or anomaly detection (Angelopoulos et al., 2019; Md et al., 2022), which allows prompt corrective actions to optimize energy usage and prevent energy inefficiencies. Within this context, ML techniques for image analysis (Casini et al., 2024) are also gaining increasing interest (Chen et al., 2023), for their application to e.g. materials design and optimization (Choudhury, 2021), quality control (Badmos et al., 2020), process monitoring (Ho et al., 2021), or detection of machine failures by converting time series data from sensors to 2D images (Wen et al., 2017).

Incorporating digitalisation and ML techniques into Industry 4.0 has led to significant energy savings (Maggiore et al., 2021; Nota et al., 2020). Projects adopting these technologies can achieve an average of 15% to 25% improvement in energy efficiency in the processes where they were implemented (Arana-Landín et al., 2023). For instance, in predictive maintenance, ML can reduce energy consumption by optimizing the operation of machinery (Agrawal et al., 2023; Pan et al., 2024). In process optimization, ML algorithms can improve energy efficiency by 10–20% by analyzing and adjusting machine operations for optimal performance, thereby reducing unnecessary energy usage (Leong et al., 2020). Furthermore, the implementation of ML algorithms for optimal control can lead to energy savings of 30%, because these systems can make real-time adjustments to production lines, ensuring that machines operate at peak energy efficiency (Rahul & Chiddarwar, 2023).

In automotive manufacturing, ML-driven quality control can lead to energy savings by reducing the need for redoing parts or running inefficient production cycles (Vater et al., 2019). In high-volume production environments such as consumer electronics,

novel computer-based vision models for automated detection and classification of damaged packages from intact packages can speed up operations and reduce waste (Shahin et al., 2023). In heavy industries like steel or chemical manufacturing, ML can optimize the energy consumption of large machinery. By predicting the optimal operating conditions and maintenance schedules, these systems can save energy costs (Mypati et al., 2023). Compressed air is one of the most energy-intensive processes in manufacturing. ML can optimize the performance of these systems, potentially leading to energy savings by continuously monitoring and adjusting the air compressors for peak efficiency, avoiding energy losses due to leaks or inefficient operation (Benedetti et al., 2019). ML can also contribute to reducing energy consumption and minimizing incorrectly produced parts in polymer processing enterprises (Willenbacher et al., 2021).

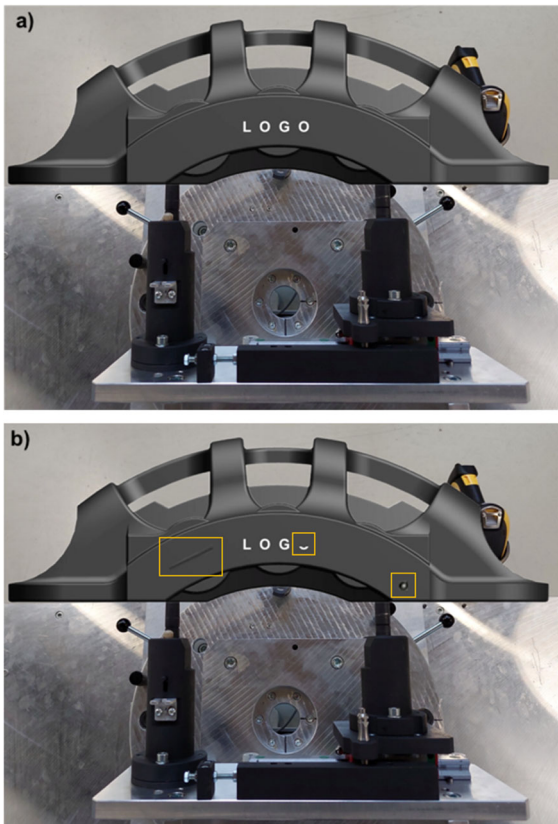
Here we focus on a practical industrial case study of brake caliper processing. In detail, we focus on the quality control operation, which is typically accomplished by human visual inspection and requires a dedicated handling system. This eventually implies a slower production rate, and inefficient energy usage. We thus propose the integration of an ML-based system to automatically perform the quality control operation, without the need for a dedicated handling system and thus reduced operation time. To this, we rely on ML tools able to analyze and extract information from images, that is, deep convolutional neural networks, D-CNNs (Alzubaidi et al., 2021; Chai et al., 2021).

A complete workflow for the purpose has been developed and tested on a real industrial test case. This includes: a dedicated pre-processing of the brake caliper images, their labelling and analysis using two dedicated D-CNN architectures (one for background removal, and one for defect identification), post-processing and analysis of the neural network output. Several different D-CNN architectures have been tested, in order to find the best model in terms of accuracy and computational demand. The results show that, a judicious choice of the ML model with a proper training, allows to obtain fast and accurate recognition of possible defects. The best-performing models, indeed, reach over 98% accuracy on the target criteria for quality control, and take only few seconds to analyze each image. These results make the proposed workflow compliant with the typical industrial expectations; therefore, in perspective, it could be implemented for an ML-

powered version of the considered industrial problem. This would eventually allow to achieve better performance of the manufacturing process and, ultimately, a better management of the available resources in terms of time consumption and energy expense.

## Case study

The industrial quality control process that we target is the visual inspection of manufactured components, to verify the absence of possible defects. Due to industrial confidentiality reasons, a representative open-source 3D geometry (GrabCAD) of the considered parts, similar to the original one, is shown in Fig. 1. For illustrative purposes, the clean geometry without defects (Fig. 1(a)) is compared to the geometry with three pos-



**Fig. 1** Sample 3D model (GrabCAD) of the considered brake caliper: (a) part without defects, and (b) part with three sample defects, namely a scratch, a partially missing letter in the logo, and a circular painting defect (shown by the yellow squares, from left to right respectively)

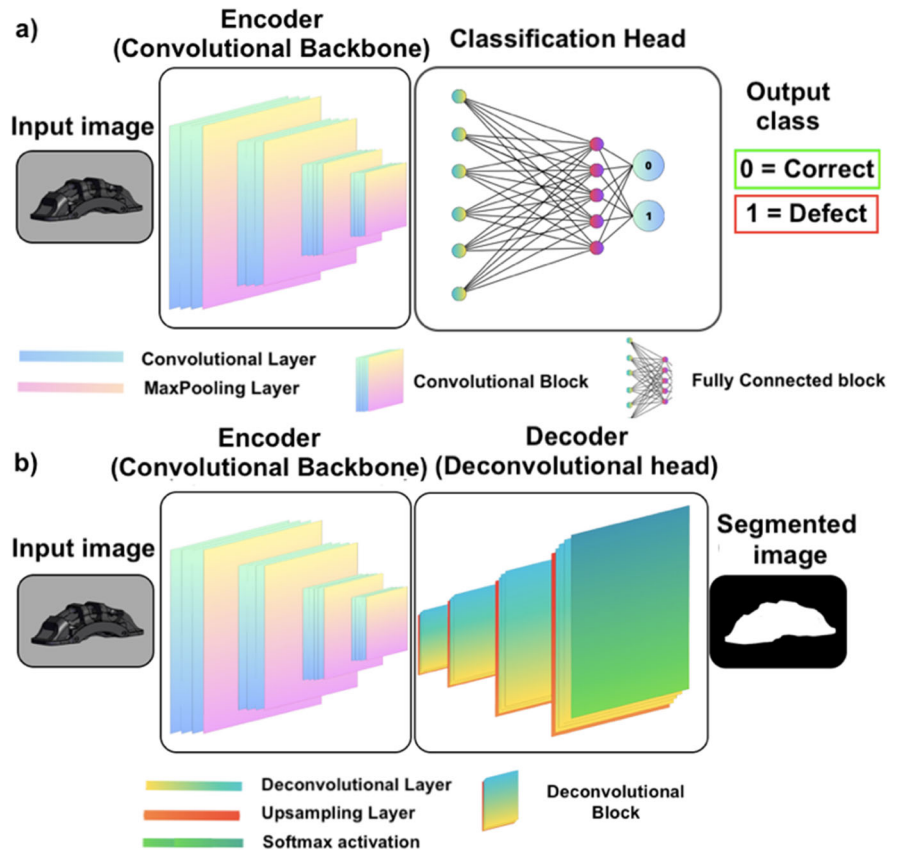
sible sample defects, namely: a scratch on the surface of the brake caliper, a partially missing letter in the logo, and a circular painting defect (highlighted by the yellow squares, from left to right respectively, in Fig. 1(b)). Note that, one or multiple defects may be present on the geometry, and that other types of defects may also be considered.

Within the industrial production line, this quality control is typically time consuming, and requires a dedicated handling system with the associated slow production rate and energy inefficiencies. Thus, we developed a methodology to achieve an ML-powered version of the control process. The method relies on data analysis and, in particular, on information extraction from images of the brake calipers via Deep Convolutional Neural Networks, D-CNNs (Alzubaidi et al., 2021). The designed workflow for defect recognition is implemented in the following two steps: 1) removal of the background from the image of the caliper, in order to reduce noise and irrelevant features in the image, ultimately rendering the algorithms more flexible with respect to the background environment; 2) analysis of the geometry of the caliper to identify the different possible defects. These two serial steps are accomplished via two different and dedicated neural networks, whose architecture is discussed in the next section.

## Methods

Convolutional Neural Networks (CNNs) pertain to a particular class of deep neural networks for information extraction from images. The feature extraction is accomplished via convolution operations; thus, the algorithms receive an image as an input, analyze it across several (deep) neural layers to identify target features, and provide the obtained information as an output (Casini et al., 2024). Regarding this latter output, different formats can be retrieved based on the considered architecture of the neural network. For a numerical data output, such as that required to obtain a classification of the content of an image (Bhatt et al., 2021), e.g. correct or defective caliper in our case, a typical layout of the network involving a convolutional backbone, and a fully-connected network can be adopted (see Fig. 2(a)). On the other hand, if the required output is still an image, a more complex architecture with a convolutional backbone (encoder) and a deconvolutional head (decoder) can be used (see Fig. 2(b)).

**Fig. 2** Different neural network architectures: convolutional encoder (a) and encoder-decoder (b)



As previously introduced, our workflow targets the analysis of the brake calipers in a two-step procedure: first, the removal of the background from the input image (e.g. Fig. 1); second, the geometry of the caliper is analyzed and the part is classified as acceptable or not depending on the absence or presence of any defect, respectively. Thus, in the first step of the procedure, a dedicated encoder-decoder network (Minaee et al., 2021) is adopted to classify the pixels in the input image as brake or background. The output of this model will then be a new version of the input image, where the background pixels are blacked. This helps the algorithms in the subsequent analysis to achieve a better performance, and to avoid bias due to possible different environments in the input image. In the second step of the workflow, a dedicated encoder architecture is adopted. Here, the previous background-filtered image is fed to the convolutional network, and the geometry of the caliper is analyzed to spot possible defects and thus classify the part as acceptable or not. In this work, both deep learning models are *supervised*, that is, the algorithms are trained with the help of human-

labeled data (LeCun et al., 2015). Particularly, the first algorithm for background removal is fed with the original image as well as with a *ground truth* (i.e. a binary image, also called *mask*, consisting of black and white pixels) which instructs the algorithm to learn which pixels pertain to the brake and which to the background. This latter task is usually called *semantic segmentation* in Machine Learning and Deep Learning (Géron, 2022). Analogously, the second algorithm is fed with the original image (without the background) along with an associated mask, which serves the neural networks with proper instructions to identify possible defects on the target geometry. The required pre-processing of the input images, as well as their use for training and validation of the developed algorithms, are explained in the next sections.

### Image pre-processing

Machine Learning approaches rely on data analysis; thus, the quality of the final results is well known to depend strongly on the amount and quality of the avail-

able data for training of the algorithms (Banko & Brill, 2001; Chen et al., 2021). In our case, the input images should be well-representative for the target analysis and include adequate variability of the possible features to allow the neural networks to produce the correct output. In this view, the original images should include, e.g., different possible backgrounds, a different viewing angle of the considered geometry and a different light exposure (as local light reflections may affect the color of the geometry and thus the analysis). The creation of such a proper dataset for specific cases is not always straightforward; in our case, for example, it would imply a systematic acquisition of a large set of images in many different conditions. This would require, in turn, disposing of all the possible target defects on the real parts, and of an automatic acquisition system, e.g., a robotic arm with an integrated camera. Given that, in our case, the initial dataset could not be generated on real parts, we have chosen to generate a well-balanced dataset of images *in silico*, that is, based on image renderings of the real geometry. The key idea was that, if the rendered geometry is sufficiently close to a real photograph, the algorithms may be instructed on artificially-generated images and then tested on a few real ones. This approach, if properly automatized, clearly allows to easily produce a large amount of images in all the different conditions required for the analysis.

In a first step, starting from the CAD file of the brake calipers, we worked manually using the open-source software Blender (Blender), to modify the material properties and achieve a realistic rendering. After that, defects were generated by means of Boolean (subtraction) operations between the geometry of the brake caliper and *ad-hoc* geometries for each defect. Fine tuning on the generated defects has allowed for a realistic representation of the different defects. Once the results were satisfactory, we developed an automated Python code for the procedures, to generate the renderings in different conditions. The Python code allows to: load a given CAD geometry, change the material properties, set different viewing angles for the geometry, add different types of defects (with given size, rotation and location on the geometry of the brake caliper), add a custom background, change the lighting conditions, render the scene and save it as an image.

In order to make the dataset as varied as possible, we introduced three light sources into the rendering environment: a diffuse natural lighting to simulate day-

light conditions, and two additional artificial lights. The intensity of each light source and the viewing angle were then made vary randomly, to mimic different day-light conditions and illuminations of the object. This procedure was designed to provide different situations akin to real use, and to make the model invariant to lighting conditions and camera position. Moreover, to provide additional flexibility to the model, the training dataset of images was virtually expanded using data augmentation (Mumuni & Mumuni, 2022), where saturation, brightness and contrast were made randomly vary during training operations. This procedure has allowed to consistently increase the number and variety of the images in the training dataset.

The developed automated pre-processing steps easily allows for batch generation of thousands of different images to be used for training of the neural networks. This possibility is key for proper training of the neural networks, as the variability of the input images allows the models to learn all the possible features and details that may change during real operating conditions.

The first tests using such virtual database have shown that, although the generated images were very similar to real photographs, the models were not able to properly recognize the target features in the real images. Thus, in a tentative to get closer to a proper set of real images, we decided to adopt a hybrid dataset, where the virtually generated images were mixed with the available few real ones. However, given that some possible defects were missing in the real images, we also decided to manipulate the images to introduce virtual defects on real images. The obtained dataset finally included more than 4,000 images, where 90% was rendered, and 10% was obtained from real images. To avoid possible bias in the training dataset, defects were present in 50% of the cases in both the rendered and real image sets. Thus, in the overall dataset, the real original images with no defects were 5% of the total.

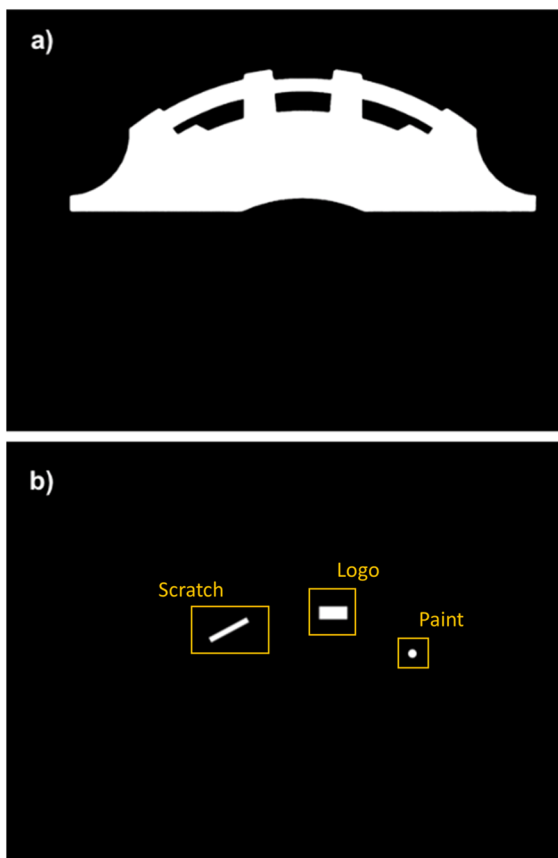
Along with the code for the rendering and manipulation of the images, dedicated Python routines were developed to generate the corresponding data labelling for the supervised training of the networks, namely the image masks. Particularly, two masks were generated for each input image: one for the background removal operation, and one for the defect identification. In both cases, the masks consist of a binary (i.e. black and white) image where all the pixels of a target feature (i.e. the geometry or defect) are assigned unitary values (white); whereas, all the remaining pixels are blacked

(zero values). An example of these masks in relation to the geometry in Fig. 1 is shown in Fig. 3.

All the generated images were then down-sampled, that is, their resolution was reduced to avoid unnecessary large computational times and (RAM) memory usage while maintaining the required level of detail for training of the neural networks. Finally, the input images and the related masks were split into a *mosaic* of smaller tiles, to achieve a suitable size for feeding the images to the neural networks with even more reduced requirements on the RAM memory. All the tiles were processed, and the whole image reconstructed at the end of the process to visualize the overall final results.

### Choice of the model

Within the scope of the present application, a wide range of possibly suitable models is available (Chen



**Fig. 3** Examples of the ground truth for the two target tasks: background removal (a) and defects recognition (b)

et al., 2021). In general, the choice of the best model for a given problem should be made on a case-by-case basis, considering an acceptable compromise between the achievable accuracy and computational complexity/cost. Too simple models can indeed be very fast in the response yet have a reduced accuracy. On the other hand, more complex models can generally provide more accurate results, although typically requiring larger amounts of data for training, and thus longer computational times and energy expense. Hence, testing has the crucial role to allow identification of the best trade-off between these two extreme cases. A benchmark for model accuracy can generally be defined in terms of a confusion matrix, where the model response is summarized into the following possibilities: True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN). This concept can be summarized as shown in Fig. 4. For the background removal, Positive (P) stands for pixels belonging to the brake caliper, while Negative (N) for background pixels. For the defect identification model, Positive (P) stands for non-defective geometry, whereas Negative (N) stands for defective geometries. With respect to these two cases, the True/False statements stand for correct or incorrect identification, respectively. The model accuracy can be therefore assessed as Géron (2022)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \quad (1)$$

Based on this metrics, the accuracy for different models can then be evaluated on a given dataset, where typically 80% of the data is used for training and the remaining 20% for validation. For the defect recognition stage, the following models were tested: VGG-16 (Simonyan & Zisserman, 2014), ResNet50, ResNet101, ResNet152 (He et al., 2016), Inception V1 (Szegedy et al., 2015), Inception V4 and InceptionResNet V2 (Szegedy et al., 2017). Details on the assessment procedure for the different models are provided in the Supplementary Information file. For the background removal stage, the DeepLabV3+ (Chen et al., 2018) model was chosen as the first option, and no additional models were tested as it directly provided satisfactory results in terms of accuracy and processing time. This gives preliminary indication that, from the point of view of the task complexity of the problem, the defect identification stage can be more demanding with respect to the background removal operation for the

**Fig. 4** Confusion matrix for accuracy assessment of the neural networks models

		Predicted	
		Positive (P)	Negative (N)
Actual	Total population = P + N		
	Positive (P)	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)

case study at hand. Besides the assessment of the accuracy according to, e.g., the metrics discussed above, additional information can be generally collected, such as too low accuracy (indicating insufficient amount of training data), possible bias of the models on the data (indicating a non-well balanced training dataset), or other specific issues related to missing representative data in the training dataset (Géron, 2022). This information helps both to correctly shape the training dataset, and to gather useful indications for the fine tuning of the model after its choice has been made.

## Results

### Background removal

An initial bias of the model for background removal arose on the color of the original target geometry (red color). The model was indeed identifying possible red spots on the background as part of the target geometry as an unwanted output. To improve the model flexibility, and thus its accuracy on the identification of the background, the training dataset was expanded using data augmentation (Géron, 2022). This technique allows to artificially increase the size of the training dataset by applying various transformations to the available images, with the goal to improve the performance and generalization ability of the models. This approach typically involves applying geometric and/or color transformations to the original images; in our case, to account for different viewing angles of the geometry, different light exposures, and different color reflections and shadowing effects. These improvements of the training dataset proved to be effective on the performance for the background removal operation, with a validation accuracy finally ranging above 99% and model response time around 1-2 seconds. An example

of the output of this operation for the geometry in Fig. 1 is shown in Fig. 5.

While the results obtained were satisfactory for the original (red) color of the calipers, we decided to test the model ability to be applied on brake calipers of other colors as well. To this, the model was trained and tested on a grayscale version of the images of the calipers, which allows to completely remove any possible bias of the model on a specific color. In this case, the validation accuracy of the model was still obtained to range above 99%; thus, this approach was found to be particularly interesting to make the model suitable for background removal operation even on images including calipers of different colors.

### Defect recognition

An overview of the performance of the tested models for the defect recognition operation on the original geometry of the caliper is reported in Table 1 (see also the Supplementary Information file for more details on



**Fig. 5** Target geometry after background removal

**Table 1** Comparison of the models for defect recognition on our dataset

Model	$A_v$ [%]	$N_p$ ( $\times 10^6$ )	Notes
VGG-16	62.5	138	Unprecise
ResNet50	92.8	26	Good
ResNet101	94.6	44,6	Best choice
ResNet152	94.8	230	Slow and complex
Inception v1	91.6	5	Good
Inception v4	94.5	43	Very good
Inception-Resnet v2	94.7	55	Very good but slow

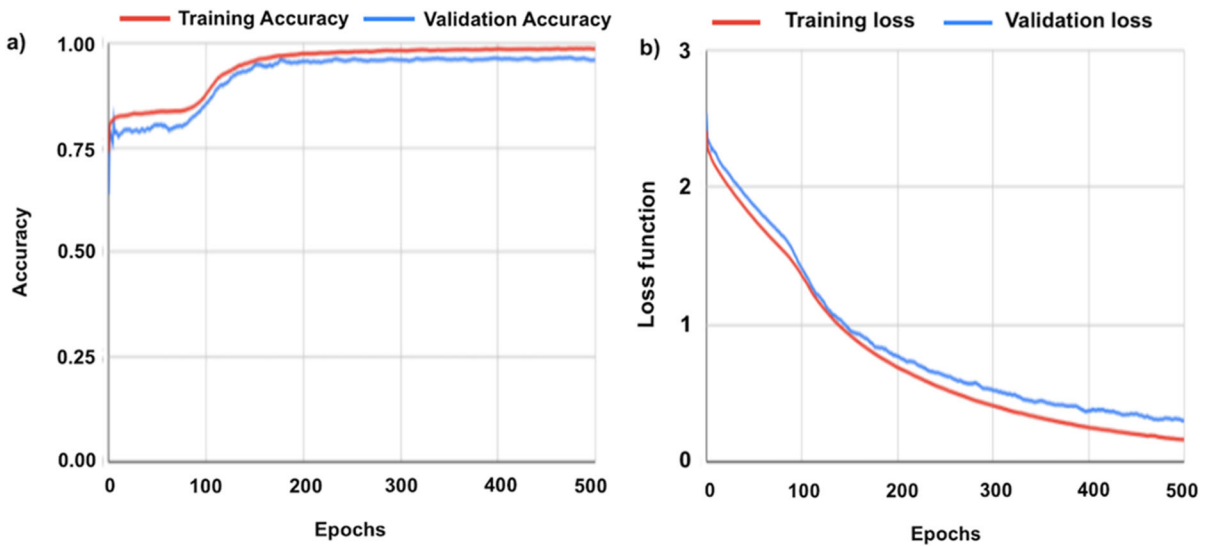
the assessment of different models). The results report on the achieved validation accuracy ( $A_v$ ) and on the number of parameters ( $N_p$ ), with this latter being the total number of parameters that can be trained for each model (Géron, 2022) to determine the output. Here, this quantity is adopted as an indicator of the complexity of each model.

As the results in Table 1 show, the VGG-16 model was quite unprecise for our dataset, eventually showing underfitting (Géron, 2022). Thus, we decided to opt for the Resnet and Inception families of models. Both these families of models have demonstrated to be suitable for handling our dataset, with slightly less accurate results being provided by the Resnet50 and InceptionV1. The best results were obtained using Resnet101 and InceptionV4, with very high final accuracy and fast processing time (in the order  $\sim 1$  second). Finally, Resnet152 and InceptionResnetV2 models proved to be slightly too complex or slower for our case; they indeed provided excellent results but taking longer response times (in the order of  $\sim 3$ -5 seconds). The response time is indeed affected by the complexity ( $N_p$ ) of the model itself, and by the hardware used. In our work, GPUs were used for training and testing all the models, and the hardware conditions were kept the same for all models.

Based on the results obtained, ResNet101 model was chosen as the best solution for our application, in terms of accuracy and reduced complexity. After fine-tuning operations, the accuracy that we obtained with this model reached nearly 99%, both in the validation and test datasets. This latter includes target real images, that the models have never seen before; thus, it can be used for testing of the ability of the models to generalize the information learnt during the training/validation phase.

The trend in the accuracy increase and loss function decrease during training of the Resnet101 model on the original geometry are shown in Fig. 6(a) and (b), respectively. Particularly, the loss function quantifies the error between the predicted output during training of the model and the actual target values in the dataset. In our case, the loss function is computed using the cross-entropy function and the Adam optimiser (Géron, 2022). The error is expected to reduce during the training, which eventually leads to more accurate predictions of the model on previously-unseen data. The combination of accuracy and loss function trends, along with other control parameters, is typically used and monitored to evaluate the training process, and avoid e.g. under- or over-fitting problems (Géron, 2022). As Fig. 6(a) shows, the accuracy experiences a sudden step increase during the very first training phase (epochs, that is, the number of times the complete database is repeatedly scrutinized by the model during its training (Géron, 2022)). The accuracy then increases in a smooth fashion with the epochs, until an asymptotic value is reached both for training and validation accuracy. These trends in the two accuracy curves can generally be associated with a proper training; indeed, being the two curves close to each other may be interpreted as an absence of under-fitting problems. On the other hand, Fig. 6(b) shows that the loss function curves are close to each other, with a monotonically-decreasing trend. This can be interpreted as an absence of over-fitting problems, and thus of proper training of the model.

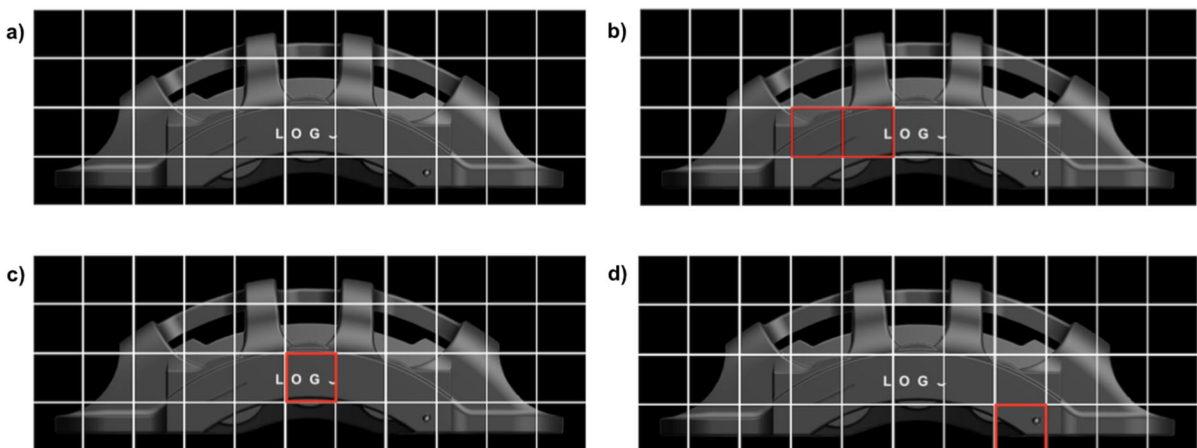
Finally, an example output of the overall analysis is shown in Fig. 7, where the considered input geometry is shown (a), along with the identification of the defects (b), (c) and (d) obtained from the developed protocol. Note that, here the different defects have been sepa-



**Fig. 6** Accuracy (a) and loss function (b) curves for the Resnet101 model during training

rated in several figures for illustrative purposes; however, the analysis yields the identification of defects on one single image. In this work, a binary classification was performed on the considered brake calipers, where the output of the models allows to discriminate between defective or non-defective components based on the presence or absence of any of the considered defects. Note that, fine tuning of this discrimination is ultimately with the user's requirements. Indeed, the model output yields as the probability (from 0 to 100%) of the possible presence of defects; thus, the discrimi-

nation between a defective or non-defective part is ultimately with the user's choice of the acceptance threshold for the considered part (50% in our case). Therefore, stricter or looser criteria can be readily adopted. Eventually, for particularly complex cases, multiple models may also be used concurrently for the same task, and the final output defined based on a cross-comparison of the results from different models. As a last remark on the proposed procedure, note that here we adopted a binary classification based on the presence or absence of any defect; however, further classification of the dif-



**Fig. 7** Final results of the analysis on the defect identification: (a) considered input geometry, (b), (c) and (d) identification of a scratch on the surface, partially missing logo, and painting defect respectively (highlighted in the red frames)

ferent defects could also be implemented, to distinguish among different types of defects (multi-class classification) on the brake calipers.

## Energy saving

### Illustrative scenarios

Given that the proposed tools have not yet been implemented and tested within a real industrial production line, we analyze here three perspective scenarios to provide a practical example of the potential for energy savings in an industrial context. To this, we consider three scenarios, which compare traditional human-based control operations and a quality control system enhanced by the proposed Machine Learning (ML) tools. Specifically, here we analyze a generic brake caliper assembly line formed by 14 stations, as outlined in Table 1 in the work by Burduk and Górnicka (2017). This assembly line features a critical inspection station dedicated to defect detection, around which we construct three distinct scenarios to evaluate the efficacy of traditional human-based control operations versus a quality control system augmented by the proposed ML-based tools, namely:

*First Scenario (S1): Human-Based Inspection.* The traditional approach involves a human operator responsible for the inspection tasks.

*Second Scenario (S2): Hybrid Inspection.* This scenario introduces a hybrid inspection system where our proposed ML-based automatic detection tool assists the human inspector. The ML tool analyzes the brake calipers and alerts the human inspector only when it encounters difficulties in identifying defects, specifically when the probability of a defect being present or absent falls below a certain threshold. This collaborative approach aims to combine the precision of ML algorithms with the experience of human inspectors, and can be seen as a possible transition scenario between the human-based and a fully-automated quality control operation.

*Third Scenario (S3): Fully Automated Inspection.* In the final scenario, we conceive a completely automated defect inspection station powered exclusively by our ML-based detection system. This setup eliminates the need for human intervention, relying entirely on the capabilities of the ML tools to identify defects.

For simplicity, we assume that all the stations are aligned in series without buffers, minimizing unneces-

sary complications in our estimations. To quantify the beneficial effects of implementing ML-based quality control, we adopt the Overall Equipment Effectiveness (OEE) as the primary metric for the analysis. OEE is a comprehensive measure derived from the product of three critical factors, as outlined by Nota et al. (2020): *Availability* (the ratio of operating time with respect to planned production time); *Performance* (the ratio of actual output with respect to the theoretical maximum output); and *Quality* (the ratio of the good units with respect to the total units produced). In this section, we will discuss the details of how we calculate each of these factors for the various scenarios.

To calculate Availability ( $A$ ), we consider an 8-hour work shift ( $t_{shift}$ ) with 30 minutes of breaks ( $t_{break}$ ) during which we assume production stop (except for the fully automated scenario), and 30 minutes of scheduled downtime ( $t_{sched}$ ) required for machine cleaning and startup procedures. For unscheduled downtime ( $t_{unsched}$ ), primarily due to machine breakdowns, we assume an average breakdown probability ( $\rho_{down}$ ) of 5% for each machine, with an average repair time of one hour per incident ( $t_{down}$ ). Based on these assumptions, since the Availability represents the ratio of run time ( $t_{run}$ ) to production time ( $t_{pt}$ ), it can be calculated using the following formula:

$$A = \frac{t_{run}}{t_{pt}} = \frac{t_{shift} - t_{break} - t_{sched} - t_{unsched}}{t_{shift} - t_{break} - t_{sched}} \quad (2)$$

with the unscheduled downtime being computed as follows:

$$t_{unsched} = t_{down} \left(1 - (1 - \rho_{down})^N\right) \quad (3)$$

where  $N$  is the number of machines in the production line and  $1 - (1 - \rho_{down})^N$  represents the probability that at least one machine breaks during the work shift. For the sake of simplicity, the  $t_{down}$  is assumed constant regardless of the number of failures.

Table 2 presents the numerical values used to calculate Availability in the three scenarios. In the second scenario, we can observe that integrating the automated station leads to a decrease in the first factor of the OEE analysis, which can be attributed to the additional station for automated quality-control (and the related potential failure). This ultimately increases the

**Table 2** Comparison of Availability across the three scenarios (S1: Human-Based Inspection, S2: Hybrid Inspection, S3: Fully Automated Inspection)

	S1	S2	S3
$t_{shift}$ [min]	480	480	480
$t_{break}$ [min]	30	30	0
$t_{sched}$ [min]	30	30	30
$t_{unsched}$ [min]	29.2	30.7	30.7
$N$ [-]	13	14	14
<b>A</b> [%]	93.05	92.68	93.17

estimation of the unscheduled downtime. In the third scenario, the detrimental effect of the additional station compensates the beneficial effect of the automated quality control on reducing the need for pauses during operator breaks; thus, the Availability for the third scenario yields as substantially equivalent to the first one (baseline).

The second factor of OEE, Performance ( $P$ ), assesses the operational efficiency of production equipment relative to its maximum designed speed ( $t_{line}$ ). This evaluation includes accounting for reductions in cycle speed and minor stoppages, collectively termed as *speed losses*. These losses are challenging to measure in advance, as performance is typically measured using historical data from the production line. For this analysis, we are constrained to hypothesize a reasonable estimate of 60 seconds of time lost to speed losses ( $t_{losses}$ ) for each work cycle. Although this assumption may appear strong, it will become evident later that, within the context of this analysis – particularly regarding the impact of automated inspection on energy savings – the Performance (like the Availability) is only marginally influenced by introducing an automated inspection station. To account for the effect of automated inspection on the assembly line speed, we keep the time required by the other 13 stations ( $t_{line}^*$ ) constant while varying the time allocated for visual inspection ( $t_{inspect}$ ). According to Burduk and Górnicka (2017), the total operation time of the production line, excluding inspection, is 1263 seconds, with manual visual inspection taking 38 seconds. For the fully automated third scenario, we assume an inspection time of 5 seconds, which encloses the photo collection, pre-processing, ML-analysis, and post-processing steps. In the second scenario, instead, we add an additional time to the pure automatic case to consider the cases when the confi-

dence of the ML model falls below 90%. We assume this happens once in every 10 inspections, which is a conservative estimate, higher than that we observed during model testing. This results in adding 10% of the human inspection time to the fully automated time. Thus, when  $t_{losses}$  are known, Performance can be expressed as follows:

$$P = \frac{t_{line}}{t_{line} + t_{losses}} = \frac{t_{line}^* + t_{inspect}}{t_{line}^* + t_{inspect} + t_{losses}} \quad (4)$$

The calculated values for Performance are presented in Table 3, and we can note that the modification in inspection time has a negligible impact on this factor since it does not affect the speed loss or, at least to our knowledge, there is no clear evidence to suggest that the introduction of a new inspection station would alter these losses. Moreover, given the specific linear layout of the considered production line, the inspection time change has only a marginal effect on enhancing the production speed. However, this approach could potentially bias our scenario towards always favouring automation. To evaluate this hypothesis, a sensitivity analysis which explores scenarios where the production line operates at a faster pace will be discussed in the next subsection.

The last factor, Quality ( $Q$ ), quantifies the ratio of compliant products out of the total products manufactured, effectively filtering out items that fail to meet the quality standards due to defects. Given the objective of our automated algorithm, we anticipate this factor of the OEE to be significantly enhanced by implementing the ML-based automated inspection station. To estimate it, we assume a constant defect probability for the production line ( $\rho_{def}$ ) at 5%. Consequently, the num-

**Table 3** Comparison of Performance across the three scenarios (S1: Human-Based Inspection, S2: Hybrid Inspection, S3: Fully Automated Inspection)

	S1	S2	S3
$t_{line}^*$ [s]	1263	1263	1263
$t_{inspect}$ [s]	38	8,8	5
$t_{losses}$ [s]	60	60	60
<b>P</b> [%]	95.59	95.49	95.48

ber of defective products ( $N_{def}$ ) during the work shift is calculated as  $N_{unit} \cdot \rho_{def}$ , where  $N_{unit}$  represents the average number of units (brake calipers) assembled on the production line, defined as:

$$N_{unit} = \frac{t_{shift} - t_{break} - t_{sched} - t_{unsched}}{t_{line}^* + t_{inspect} + t_{losses}} \tag{5}$$

To quantify defective units identified, we consider the inspection accuracy ( $\rho_{acc}$ ), where for human visual inspection, the typical accuracy is 80% (Sundaram & Zeid, 2023), and for the ML-based station, we use the accuracy of our best model, i.e., 99%. Additionally, we account for the probability of the station mistakenly identifying a caliper as with a defect even if it is defect-free, i.e., the false negative rate ( $\rho_{FN}$ ), defined as

$$\rho_{FN} = \frac{FN}{FN + FP} \tag{6}$$

In the absence of any reasonable evidence to justify a bias on one mistake over others, we assume a uniform distribution for both human and automated inspections regarding error preference, i.e. we set  $\rho_{FN}^H = \rho_{FN}^{ML} = \rho_{FN} = 50\%$ . Thus, the number of final compliant goods ( $N_{goods}$ ), i.e., the calipers that are identified as quality-compliant, can be calculated as:

$$\begin{aligned} N_{goods} &= N_{unit} - N_{detect} \tag{7} \\ &= N_{unit} - TN - FN \\ &= N_{unit} - N_{unit} \cdot \rho_{def} \cdot \rho_{acc} - \\ &\quad N_{unit} \cdot (1 - \rho_{def}) \cdot (1 - \rho_{acc}) \cdot \rho_{FN}, \end{aligned}$$

where  $N_{detect}$  is the total number of detected defective units, comprising  $TN$  (true negatives, i.e. correctly identified defective calipers) and  $FN$  (false negatives, i.e. calipers mistakenly identified as defect-free). The Quality factor can then be computed as:

$$Q = \frac{N_{goods}}{N_{unit}} = \frac{N_{unit} - N_{detect}}{N_{unit}} \tag{8}$$

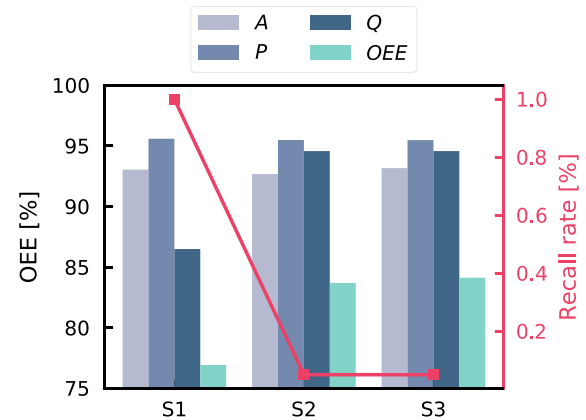
Table 4 summarizes the Quality factor calculation, showcasing the substantial improvement brought by the ML-based inspection station due to its higher accuracy compared to human operators.

Finally, we can determine the Overall Equipment Effectiveness by multiplying the three factors previ-

**Table 4** Comparison of Quality across the three scenarios (S1: Human-Based Inspection, S2: Hybrid Inspection, S3: Fully Automated Inspection)

	S1	S2	S3
$N_{unit}$ [-]	17.23	17.53	18.94
$N_{def}$ [-]	0.86	0.88	0.88
$TN$ [-]	0.89	0.87	0.87
$FN$ [-]	1.64	0.09	0.09
$N_{detect}$ [-]	2.33	0.95	0.95
<b>Q</b> [%]	86.50	94.58	94.58

ously computed. Additionally, we can estimate the recall rate ( $\rho_R$ ), which reflects the rate at which a customer might reject products. This is derived from the difference between the total number of defective units,  $N_{def}$ , and the number of units correctly identified as defective,  $TN$ , indicating the potential for defective brake calipers that may bypass the inspection process. In Fig. 8 we summarize the outcomes of the three scenarios. It is crucial to note that the scenarios incorporating the automated defect detector, S2 and S3, significantly enhance the Overall Equipment Effectiveness, primarily through substantial improvements in the Quality factor. Among these, the fully automated inspection scenario, S3, emerges as a slightly supe-



**Fig. 8** Overall Equipment Effectiveness (OEE) analysis for three scenarios (S1: Human-Based Inspection, S2: Hybrid Inspection, S3: Fully Automated Inspection). The height of the bars represents the percentage of the three factors  $A$ : Availability,  $P$ : Performance, and  $Q$ : Quality, which can be interpreted from the left axis. The green bars indicate the OEE value, derived from the product of these three factors. The red line shows the recall rate, i.e. the probability that a defective product is rejected by the client, with values displayed on the right red axis

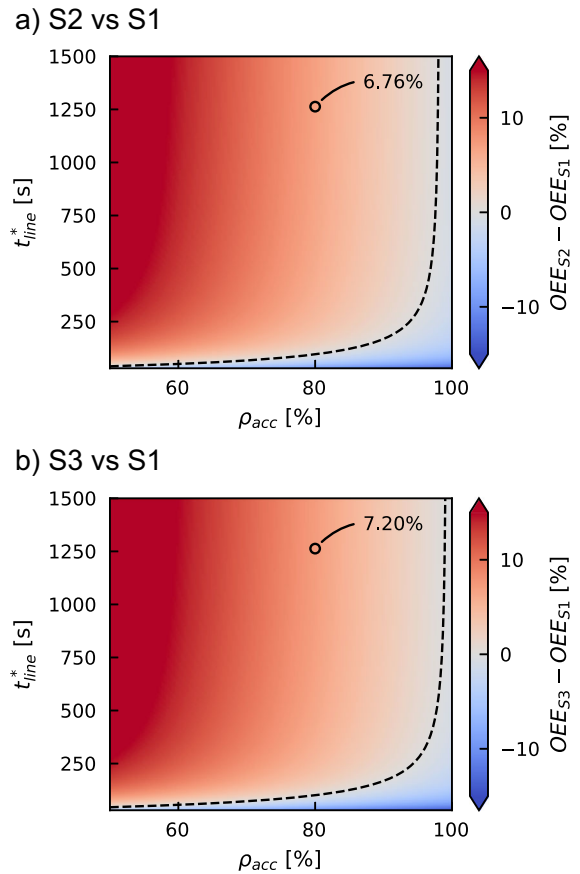
rior option, thanks to its additional benefit in removing the breaks and increasing the speed of the line. However, given the different assumptions required for this OEE study, we shall interpret these results as illustrative, and considering them primarily as comparative with the baseline scenario only. To analyze the sensitivity of the outlined scenarios on the adopted assumptions, we investigate the influence of the line speed and human accuracy on the results in the next subsection.

### Sensitivity analysis

The scenarios described previously are illustrative and based on several simplifying hypotheses. One of such hypotheses is that the production chain layout operates entirely in series, with each station awaiting the arrival of the workpiece from the preceding station, resulting in a relatively slow production rate (1263 seconds). This setup can be quite different from reality, where slower operations can be accelerated by installing additional machines in parallel to balance the workload and enhance productivity. Moreover, we utilized a literature value of 80% for the accuracy of the human visual inspector operator, as reported by Sundaram and Zeid (2023). However, this accuracy can vary significantly due to factors such as the experience of the inspector and the defect type.

A sensitivity analysis on these two factors was conducted to address these variations. The assembly time of the stations (excluding visual inspection),  $t_{line}^*$ , was varied from 60 s to 1500 s, and the human inspection accuracy,  $\rho_{acc}$ , ranged from 50% (akin to a random guesser) to 100% (representing an ideal visual inspector); meanwhile, the other variables were kept fixed.

The comparison of the OEE enhancement for the two scenarios employing ML-based inspection against the baseline scenario is displayed in the two maps in Fig. 9. As the figure shows, due to the high accuracy and rapid response of the proposed automated inspection station, the area representing regions where the process may benefit energy savings in the assembly lines (indicated in red shades) is significantly larger than the areas where its introduction could degrade performance (indicated in blue shades). However, it can be also observed that the automated inspection could be superfluous or even detrimental in those scenarios where human accuracy and assembly speed are very high, indicating an already highly accurate workflow.



**Fig. 9** Effect of assembly time for stations (excluding visual inspection),  $t_{line}^*$ , and human inspection accuracy,  $\rho_{acc}$ , on the OEE analysis. (a) The subplot shows the difference between the scenario S2 (Hybrid Inspection) and the baseline scenario S1 (Human Inspection), while subplot (b) displays the difference between scenario S3 (Fully Automated Inspection) and the baseline. The maps indicate in red the values of  $t_{line}^*$  and  $\rho_{acc}$  where the integration of automated inspection stations can significantly improve OEE, and in blue where it may lower the score. The dashed lines denote the breakeven points, and the circled points pinpoint the values of the scenarios used in the “Illustrative scenario” Subsection.

In these cases, and particularly for very fast production lines, short times for quality control can be expected to be key (beyond accuracy) for the optimization.

Finally, it is important to remark that the blue region (areas below the dashed break-even lines) might expand if the accuracy of the neural networks for defect detection is lower when implemented in a real production line. This indicates the necessity for new rounds of active learning and an augment of the ratio of real images in the database, to eventually enhance the performance of the ML model.

## Conclusions

Industrial quality control processes on manufactured parts are typically achieved by human visual inspection. This usually requires a dedicated handling system, and generally results in a slower production rate, with the associated non-optimal use of the energy resources. Based on a practical test case for quality control on brake caliper manufacturing, in this work we have reported on a developed workflow for integration of Machine Learning methods to automatize the process. The proposed approach relies on image analysis via Deep Convolutional Neural Networks. These models allow to efficiently extract information from images, thus possibly representing a valuable alternative to human inspection.

The proposed workflow relies on a two-step procedure on the images of the brake calipers: first, the background is removed from the image; second, the geometry is inspected to identify possible defects. These two steps are accomplished thanks to two dedicated neural network models, an encoder-decoder and an encoder network, respectively. Training of these neural networks typically requires a large number of representative images for the problem. Given that, one such database is not always readily available, we have presented and discussed an alternative methodology for the generation of the input database using 3D renderings. While integration of the database with real photographs was required for optimal results, this approach has allowed fast and flexible generation of a large base of representative images. The pre-processing steps required for data feeding to the neural networks and their training has been also discussed.

Several models have been tested and evaluated, and the best one for the considered case identified. The obtained accuracy for defect identification reaches  $\sim 99\%$  of the tested cases. Moreover, the response of the models is fast (in the order of few seconds) on each image, which makes them compliant with the most typical industrial expectations.

In order to provide a practical example of possible energy savings when implementing the proposed ML-based methodology for quality control, we have analyzed three perspective industrial scenarios: a baseline scenario, where quality control tasks are performed by a human inspector; a hybrid scenario, where the proposed

ML automatic detection tool assists the human inspector; a fully-automated scenario, where we envision a completely automated defect inspection. The results show that the proposed tools may help increasing the Overall Equipment Effectiveness up to  $\sim 10\%$  with respect to the considered baseline scenario. However, a sensitivity analysis on the speed of the production line and on the accuracy of the human inspector has also shown that the automated inspection could be superfluous or even detrimental in those cases where human accuracy and assembly speed are very high. In these cases, reducing the time required for quality control can be expected to be the major controlling parameter (beyond accuracy) for optimization.

Overall the results show that, with a proper tuning, these models may represent a valuable resource for integration into production lines, with positive outcomes on the overall effectiveness, and thus ultimately leading to a better use of the energy resources. To this, while the practical implementation of the proposed tools can be expected to require contained investments (e.g. a portable camera, a dedicated workstation and an operator with proper training), in field tests on a real industrial line would be required to confirm the potential of the proposed technology.

**Acknowledgements** This work has been supported by GEFIT S.p.a.

**Funding** Open access funding provided by Politecnico di Torino within the CRUI-CARE Agreement.

## Declarations

**Conflict of interest statement** The authors declare no competing interests.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Agrawal, R., Majumdar, A., Kumar, A., & Luthra, S. (2023). Integration of artificial intelligence in sustainable manufacturing: Current status and future opportunities. *Operations Management Research*, 1–22.
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, 8, 1–74.
- Angelopoulos, A., Michailidis, E. T., Nomikos, N., Trakadas, P., Hatziefremidis, A., Voliotis, S., & Zahariadis, T. (2019). Tackling faults in the industry 4.0 era—a survey of machine—learning solutions and key aspects. *Sensors*, 20(1), 109.
- Arana-Landín, G., Uriarte-Gallastegi, N., Landeta-Manzano, B., & Laskurain-Iturbe, I. (2023). The contribution of lean management—industry 4.0 technologies to improving energy efficiency. *Energies*, 16(5), 2124.
- Badmos, O., Kopp, A., Bernthaler, T., & Schneider, G. (2020). Image-based defect detection in lithium-ion battery electrode using convolutional neural networks. *Journal of Intelligent Manufacturing*, 31, 885–897. <https://doi.org/10.1007/s10845-019-01484-x>
- Banko, M., & Brill, E. (2001). Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th annual meeting of the association for computational linguistics* (pp. 26–33).
- Benedetti, M., Bonfà, F., Introna, V., Santolamazza, A., & Ubertini, S. (2019). Real time energy performance control for industrial compressed air systems: Methodology and applications. *Energies*, 12(20), 3935.
- Bhatt, D., Patel, C., Talsania, H., Patel, J., Vaghela, R., Pandya, S., Modi, K., & Ghayvat, H. (2021). Cnn variants for computer vision: History, architecture, application, challenges and future scope. *Electronics*, 10(20), 2470.
- Bilgen, S. (2014). Structure and environmental impact of global energy consumption. *Renewable and Sustainable Energy Reviews*, 38, 890–902.
- Blender. (2023). Open-source software. <https://www.blender.org/>. Accessed 18 Apr 2023.
- Bologna, A., Fasano, M., Bergamasco, L., Morciano, M., Bersani, F., Asinari, P., Meucci, L., & Chiavazzo, E. (2020). Techno-economic analysis of a solar thermal plant for large-scale water pasteurization. *Applied Sciences*, 10(14), 4771.
- Burduk, A., & Górnicka, D. (2017). Reduction of waste through reorganization of the component shipment logistics. *Research in Logistics & Production*, 7(2), 77–90. <https://doi.org/10.21008/j.2083-4950.2017.7.2.2>
- Carvalho, T. P., Soares, F. A., Vita, R., Francisco, R., d. P., Basto, J. P., & Alcalá, S. G. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering*, 137, 106024.
- Casini, M., De Angelis, P., Chiavazzo, E., & Bergamasco, L. (2024). Current trends on the use of deep learning methods for image analysis in energy applications. *Energy and AI*, 15, 100330. <https://doi.org/10.1016/j.egyai.2023.100330>
- Chai, J., Zeng, H., Li, A., & Ngai, E. W. (2021). Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications*, 6, 100134.
- Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 801–818).
- Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S., & Miao, Y. (2021). Review of image classification algorithms based on convolutional neural networks. *Remote Sensing*, 13(22), 4712.
- Chen, T., Sampath, V., May, M. C., Shan, S., Jorg, O. J., Aguilar Martín, J. J., Stamer, F., Fantoni, G., Tosello, G., & Calaon, M. (2023). Machine learning in manufacturing towards industry 4.0: From ‘for now’ to ‘four-know’. *Applied Sciences*, 13(3), 1903. <https://doi.org/10.3390/app13031903>
- Choudhury, A. (2021). The role of machine learning algorithms in materials science: A state of art review on industry 4.0. *Archives of Computational Methods in Engineering*, 28(5), 3361–3381. <https://doi.org/10.1007/s11831-020-09503-4>
- Dalzochio, J., Kunst, R., Pignaton, E., Binotto, A., Sanyal, S., Favilla, J., & Barbosa, J. (2020). Machine learning and reasoning for predictive maintenance in industry 4.0: Current status and challenges. *Computers in Industry*, 123, 103298.
- Fasano, M., Bergamasco, L., Lombardo, A., Zanini, M., Chiavazzo, E., & Asinari, P. (2019). Water/ethanol and 13x zeolite pairs for long-term thermal energy storage at ambient pressure. *Frontiers in Energy Research*, 7, 148.
- Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. O’Reilly Media, Inc.
- GrabCAD. (2023). Brake caliper 3D model by Mitulkumar Sakariya from the GrabCAD free library (non-commercial public use). <https://grabcad.com/library/brake-caliper-19>. Accessed 18 Apr 2023.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).
- Ho, S., Zhang, W., Young, W., Buchholz, M., Al Jufout, S., Dajani, K., Bian, L., & Mozumdar, M. (2021). Dlam: Deep learning based real-time porosity prediction for additive manufacturing using thermal images of the melt pool. *IEEE Access*, 9, 115100–115114. <https://doi.org/10.1109/ACCESS.2021.3105362>
- Ismail, M. I., Yunus, N. A., & Hashim, H. (2021). Integration of solar heating systems for low-temperature heat demand in food processing industry—a review. *Renewable and Sustainable Energy Reviews*, 147, 111192.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
- Leong, W. D., Teng, S. Y., How, B. S., Ngan, S. L., Abd Rahman, A., Tan, C. P., Ponnambalam, S., & Lam, H. L. (2020). Enhancing the adaptability: Lean and green strategy towards the industry revolution 4.0. *Journal of cleaner production*, 273, 122870.
- Liu, Z., Wang, X., Zhang, Q., & Huang, C. (2019). Empirical mode decomposition based hybrid ensemble model for electrical energy consumption forecasting of the cement grinding process. *Measurement*, 138, 314–324.
- Li, G., & Zheng, X. (2016). Thermal energy storage system integration forms for a sustainable future. *Renewable and Sustainable Energy Reviews*, 62, 736–757.

- Maggiore, S., Realini, A., Zagano, C., & Bazzocchi, F. (2021). Energy efficiency in industry 4.0: Assessing the potential of industry 4.0 to achieve 2030 decarbonisation targets. *International Journal of Energy Production and Management*, 6(4), 371–381.
- Mazzei, D., & Ramjattan, R. (2022). Machine learning for industry 4.0: A systematic review using deep learning-based topic modelling. *Sensors*, 22(22), 8641.
- Md, A. Q., Jha, K., Haneef, S., Sivaraman, A. K., & Tee, K. F. (2022). A review on data-driven quality prediction in the production process with machine learning for industry 4.0. *Processes*, 10(10), 1966. <https://doi.org/10.3390/pr10101966>
- Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7), 3523–3542.
- Mishra, S., Srivastava, R., Muhammad, A., Amit, A., Chiavazzo, E., Fasano, M., & Asinari, P. (2023). The impact of physicochemical features of carbon electrodes on the capacitive performance of supercapacitors: a machine learning approach. *Scientific Reports*, 13(1), 6494. <https://doi.org/10.1038/s41598-023-33524-1>
- Mumuni, A., & Mumuni, F. (2022). Data augmentation: A comprehensive survey of modern approaches. *Array*, 16, 100258. <https://doi.org/10.1016/j.array.2022.100258>
- Mypati, O., Mukherjee, A., Mishra, D., Pal, S. K., Chakrabarti, P. P., & Pal, A. (2023). A critical review on applications of artificial intelligence in manufacturing. *Artificial Intelligence Review*, 56(Suppl 1), 661–768.
- Narciso, D. A., & Martins, F. (2020). Application of machine learning tools for energy efficiency in industry: A review. *Energy Reports*, 6, 1181–1199.
- Nota, G., Nota, F. D., Peluso, D., & Toro Lazo, A. (2020). Energy efficiency in industry 4.0: The case of batch production processes. *Sustainability*, 12(16), 6631. <https://doi.org/10.3390/su12166631>
- Ocampo-Martinez, C., et al. (2019). Energy efficiency in discrete-manufacturing systems: Insights, trends, and control strategies. *Journal of Manufacturing Systems*, 52, 131–145.
- Pan, Y., Hao, L., He, J., Ding, K., Yu, Q., & Wang, Y. (2024). Deep convolutional neural network based on self-distillation for tool wear recognition. *Engineering Applications of Artificial Intelligence*, 132, 107851.
- Qin, J., Liu, Y., Grosvenor, R., Lacan, F., & Jiang, Z. (2020). Deep learning-driven particle swarm optimisation for additive manufacturing energy optimisation. *Journal of Cleaner Production*, 245, 118702.
- Rahul, M., & Chiddarwar, S. S. (2023). Integrating virtual twin and deep neural networks for efficient and energy-aware robotic deburring in industry 4.0. *International Journal of Precision Engineering and Manufacturing*, 24(9), 1517–1534.
- Ribezzo, A., Falciani, G., Bergamasco, L., Fasano, M., & Chiavazzo, E. (2022). An overview on the use of additives and preparation procedure in phase change materials for thermal energy storage with a focus on long term applications. *Journal of Energy Storage*, 53, 105140.
- Shahin, M., Chen, F. F., Hosseinzadeh, A., Bouzary, H., & Shahin, A. (2023). Waste reduction via image classification algorithms: Beyond the human eye with an ai-based vision. *International Journal of Production Research*, 1–19.
- Shen, F., Zhao, L., Du, W., Zhong, W., & Qian, F. (2020). Large-scale industrial energy systems optimization under uncertainty: A data-driven robust optimization approach. *Applied Energy*, 259, 114199.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556).
- Sundaram, S., & Zeid, A. (2023). Artificial Intelligence-Based Smart Quality Inspection for Manufacturing. *Micromachines*, 14(3), 570. <https://doi.org/10.3390/mi14030570>
- Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI conference on artificial intelligence* (vol. 31).
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1–9).
- Trezza, G., Bergamasco, L., Fasano, M., & Chiavazzo, E. (2022). Minimal crystallographic descriptors of sorption properties in hypothetical mofs and role in sequential learning optimization. *npj Computational Materials*, 8(1), 123. <https://doi.org/10.1038/s41524-022-00806-7>
- Vater, J., Schamberger, P., Knoll, A., & Winkle, D. (2019). Fault classification and correction based on convolutional neural networks exemplified by laser welding of hairpin windings. In *2019 9th International Electric Drives Production Conference (EDPC)* (pp. 1–8). IEEE.
- Wen, L., Li, X., Gao, L., & Zhang, Y. (2017). A new convolutional neural network-based data-driven fault diagnosis method. *IEEE Transactions on Industrial Electronics*, 65(7), 5990–5998. <https://doi.org/10.1109/TIE.2017.2774777>
- Willenbacher, M., Scholten, J., & Wohlgemuth, V. (2021). Machine learning for optimization of energy and plastic consumption in the production of thermoplastic parts in sme. *Sustainability*, 13(12), 6800.
- Zhang, X. H., Zhu, Q. X., He, Y. L., & Xu, Y. (2018). Energy modeling using an effective latent variable based functional link learning machine. *Energy*, 162, 883–891.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.