

Welding Defect Detection with Deep Learning Architectures

Paolo Tripicchio and Salvatore D'Avella

Abstract

Welding automation is a fundamental process in manufacturing industries. Production lines integrate welding quality controls to reduce wastes and optimize the production chain. Early detection is fundamental as defects at any stage could determine the rejection of the entire product. In the last years, following the industry 4.0 paradigm, industrial automation lines have seen the introduction of modern technologies. Although the majority of the inspection systems still rely on traditional sensing and data processing, especially in the computer vision domain, some initiatives have been taken toward the employment of machine learning architectures. This chapter introduces deep neural networks in the context of welding defect detection, starting by analyzing common problems in the industrial applications of such technologies and discussing possible solutions in the specific case of quality checks in fuel injectors welding during the production stage.

Keywords: deep learning, visual inspection, industry 4.0, welding defects, imbalanced data, transfer learning

1. Introduction

The Fourth Industrial Revolution, or Industry 4.0, aims at automating traditional manufacturing and industrial practices exploiting the most recent technologies depicted in **Figure 1**. Integrating artificial intelligence (AI) and robotics with traditional practices, the world of manufacturing processes is undergoing a transformation from activities that rely on human experience and skills into flexible environments, including objective decisional systems fully integrated within the industrial process. Advanced robotics is meant to develop autonomous and intelligent systems that could reduce the intervention of human workers [1] in many of the crucial and repetitive tasks that represent the core business of companies. Augmented and virtual reality can give operators more information about their tasks [2] and help them to alleviate mental stress during some jobs. Additive manufacturing [3] can speed up the production process. Internet of things (IoT) [4] allows new forms of communication between machines, giving rise to smart devices that can help humans achieve their objectives. Radiofrequency identification (RFID) technologies are used for efficient logistics and inventory warehouse management [5] reducing costs while increasing quality and competitiveness.

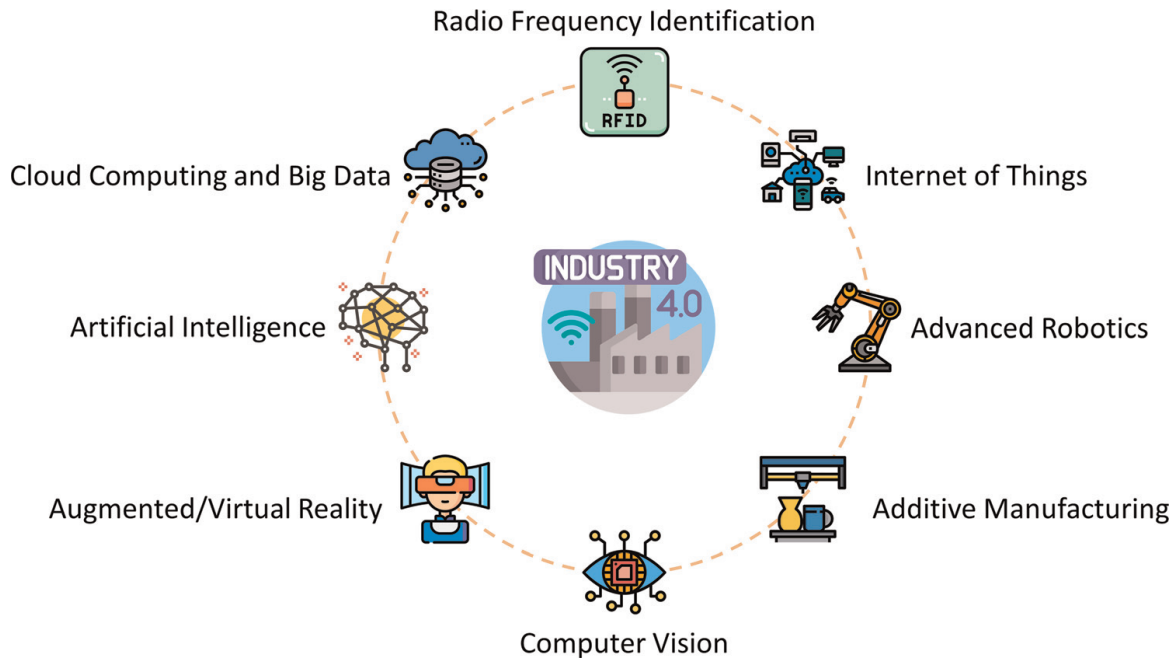


Figure 1.
Industry 4.0 pillar technologies.

Among all the aforementioned technologies, AI is perhaps the one that received more interest during the years. Indeed, nowadays, the industrial interest in AI applications in various sectors is undeniable. However, for industries, artificial intelligence is both a source of enthusiasm and skepticism. One reason is that deep learning (DL) is a technology based on data, and problems solved using AI are as good or as bad as the data they are trained on. In addition, companies perceive AI as a black box and would prefer understandable and explainable processes [6]. Both these aspects should be taken into consideration when developing industrial AI solutions.

Current automation-assisted production is mostly open-loop and relies on specific checkpoints to perform product quality analysis. Early systems based on vision date back to the nineties. Such an approach suits best when critical issues can be formally expressed by taking advantage of geometrical measurements or well-known features on the inspected objects. Unfortunately, these techniques cannot perform many quality-control activities because they need a predefined sequence of actions where quality checks should be designed carefully to meet the precise production requirements. Moreover, human nature shows formidable efficiency in learning simple checks even if it would be difficult to formalize such operations with a sequence of rules. Indeed, experience plays a relevant role in human evaluation for products quality assessment. Similarly, vision inspection processes performed by automated machines will require the development of novel algorithms that should be trained and improved with time and experience.

The introduction of automation systems in the production lines that exploit AI techniques has reduced the need for human intervention in the manufacturing process of many products. This innovation had a major impact on many industrial applications, and visual inspection is by far the activity that has profited most. Thanks to deep neural networks (DNNs), difficult computer vision tasks, such as object classification or detection and image segmentation, have been addressed recently using an adequate number of training data. DNNs are scalable, experience-based, and have similar performance to human workers. Since the development of AlexNet [7],

solutions based on deep learning have been encouraged, and convolutional neural networks (CNN) also have been extensively utilized for automating optical quality inspections. However, since such networks need a huge amount of labeled data for training their parameters, it is difficult to have an adequately large set of faulty samples with well-optimized industrial processes for creating a *balanced dataset* for efficiently training the network to defect classification. Therefore, in most cases, the objective of the training moves from defect classification to anomaly detection.

Welding is a fundamental activity in many industrial manufacturing processes, such as automotive, shipbuilding, aerospace, and electronics. It is a crucial operation for the overall quality of the production line because a defect not detected in the early stages can determine the rejection of the entire product. This chapter introduces deep neural networks in the context of welding defect detection, starting by analyzing common problems in the industrial applications of such technologies and presenting in detail a solution for quality checks in fuel injectors welding during the production stage.

2. Background

Inspection analysis can be classified into one of the following categories [8]—*structural quality*, which searches for the presence of unnecessary components or lack of required parts; *surface quality*, where objects surfaces are inspected for wear, scratches, cracks, and other defects; *dimensional quality*, where the dimensions of the objects are checked to fall within given tolerances; *operational quality*, which evaluates the correctness of the quality inspection processes.

As of today, different methods have been proposed for inspecting the welding process online [9]. Their design is suited to diverse defects types and differ in the data processed during the evaluation. Among the sensing technologies employed in literature, optical detectors [10], acoustic measurements [11], and vision analysis [12] are surely the most utilized. While, for classification applications, artificial neural networks [13–15] and fuzzy inference systems [16, 17] are usually preferred thanks to the wide range of problems and diversity of defects they could cope with as in the case of classification of steel strip defects [18, 19].

However, the focus of these works is on defects classification and not on their detection. Therefore, they could not cope with feature understanding problems such as discriminating between good samples and defective ones. A different approach is proposed by Ak et al. [20] where X-ray images are used to detect defects in metal castings.

Recent literature is plenty of research addressing the problem of welding localization employing off-the-shelf DL architectures or introducing slight modifications on the tail of popular networks. These approaches are mostly based on the R-CNN [21], Faster R-CNN [22], and YOLO [23] architectures. The reason behind their adoption is that these architectures usually require little fine-tuning procedures for efficiently localizing welding areas and spots. Such efficiency is strictly related to the presence of plain metal surfaces in the surrounding area of the welding by enabling simple and accurate segmentation of the feature under inspection. This is the case of resistance spot welding (RSW) processes typically employed to connect metal sheets at a low cost and in a short time.

Concerning detection approaches, early methods based on traditional computer vision techniques [24] require hand-crafted features and complex threshold settings

to adapt to environmental conditions. However, approaches based on deep learning allow increasing the robustness of the detection coping with environmental noise and the sensitivity of the welding processes.

The majority of approaches are built upon the above-mentioned architectures for welding spots localization. Fast R-CNN [25] is a region proposal network that computes the region of interest (ROI) on the feature map, thus improving upon the R-CNN architecture. Faster R-CNN integrates convolutional layers for object classification, feature extraction, bounding box regression, and region proposals into a network, further improving the detection performance but still not reaching real-time capabilities. Unlike the R-CNN family, which has a two-stage detection architecture, YOLO implements a regression network with a grid of bounding boxes and associated class probabilities, thus enabling real-time detection with recent hardware. In the race for timing performance, YOLOv2 [26] borrowed the anchor mechanism from SSD [27] and Faster R-CNN, which also enhanced the network *accuracy*. Focusing on small object detection (like welding spots), YOLOv3 [28] builds upon a backbone network combined with a feature pyramids network (FPN) [29] improving multi-scale prediction. The efficiency of the detection depends on the selected backbone network. Common choice are VGG [30], ResNet [31], DenseNet [32], and MobileNet [33]. These architectures differ in the computational complexity, the number of parameters, and inference speed.

Considering the reduced dimension of small spot welds, low-resolution feature maps in the backbone, and convolution strides dimension could cause an information leak. To face this issue, the work proposed by Dai et al. [34] introduces a modified MobileNetV3 [35] architecture obtaining a good tradeoff between *accuracy* and *timing*.

Focusing on the classification and detection of defects over the welding area or joint, off-the-shelf solutions are no more efficient by themselves, and some issues need to be faced to enable the use of DNNs. Clustering and segmentation become difficult because the feature to be recognized are not easily separable. This chapter introduces some of the most common issues in the employment of DL for industrial quality inspection discussing the practical case of detection of welding defects in diesel injectors heads.

3. Inspection pipeline

Quality inspection systems based on vision techniques in most cases follow the workflow depicted in **Figure 2**. The process starts by collecting the sample images using a set of cameras or sensors exploiting an adequate source of illumination. Such samples are then processed to improve images quality. Therefore, once the features are extrapolated, the evaluation of the quality and the classification of the defect are performed. Measurement and classification could either be implemented with traditional computer vision algorithms, with modern DNN architectures, or with a fusion of both of them, as in the case presented in the following. Usually, the inspection system also provides an actuation step that triggers actions, depending on the analysis result, to the production lines that directly communicate with the control unit (commonly based on programmable logic controllers (PLCs)).

The work discussed in the study by Sassi et al. [36] originated from industrial demands with the specific target of detecting welding defects on diesel injectors in the production line. Such a project focused on realizing the most effective combination of

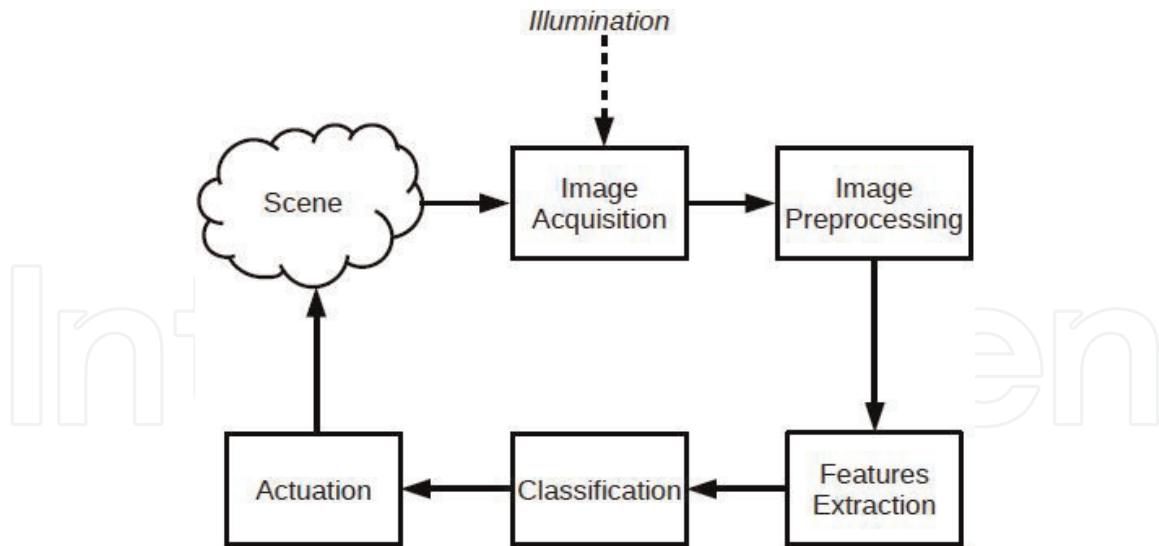


Figure 2.
Typical visual inspection workflow.

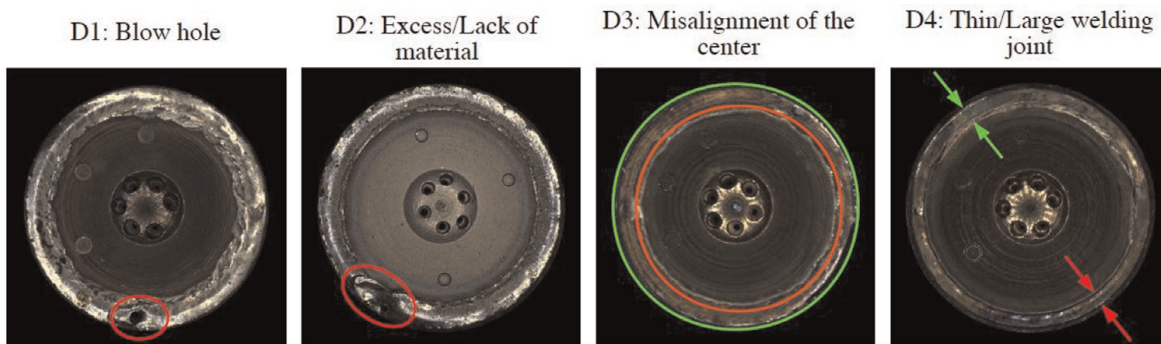


Figure 3.
Examples of defect classes. IN D3, green and red circles show the detected inner and outer edges of the welding joint. In D4, the red arrows highlight thin welding, while the green ones are standard ones.

traditional computer vision methods and deep neural network architecture for identifying the defects in the welding. In particular, the aim was to substitute the existing vision inspection system extending the classes of detectable defects in the analysis phase.

Welding joint defects may appear in different typologies: some are related to anomalies on the surface of the joint, while others are related to its geometrical properties, such as its thickness and position. Four categories have been defined for the analysis of the welding joint, as depicted graphically in **Figure 3** showing an example from each category:

- D1 (*Blowhole*): This defect corresponds to the joint area in which the material is blown up, thus, generating a loss of welding tightness and cavity;
- D2 (*Excess/Lack of material*): Blowholes are usually accompanied by a lack of material, while when they are unexploded with an excess of it;
- D3 (*Misalignment of the welding center*): When the injector head is not correctly aligned with the laser source, the resulting welding joint position is not centered;

- D4 (*Large/Thin welding joint*): When the amount of melt welding material is excessive or limited, the joint could be larger or thinner.

Defects D3 and D4 are quantitative measurable and are examined employing an algorithm based on traditional computer vision techniques (similar to the existing commercial solution). On the contrary, the others (D1 and D2) are more qualitative and are recognized through a method based on deep learning.

Furthermore, the analysis of the defects must be performed within a time slot that depends on the actual production line (1.8 seconds cycle time in the depicted scenario) to avoid interferences with the manufacturing process. This amount of time is required for the actuation system and the welding stage to process a new injector as input to the system.

4. Common challenges and possible countermeasures

During dataset preparation, the ideal case is the one in which several samples (in the order of thousands or more) are available for each class to be detected, the classes have balanced data, and they are well separated from each other. In such an ideal case, it is possible to give the network a representative set of samples of the whole input space for the training and avoid confusing the network with an uneven distribution of the inputs or the similarities between the classes.

Unfortunately, industrial production lines having well-optimized processes are usually present with few defective products and much more good samples. Therefore, it is often unfeasible to get sets of defective samples large enough to train CNNs for classification purposes. In the majority of the cases, the objective of the training moves from defect classification to anomaly detection. The worst-case scenario is the one presenting an *imbalanced dataset* with scarce availability of defect samples and classes that are not easily separable. Deep metric learning uses DNNs to directly learn a similarity metric, rather than creating it as a byproduct of solving a classification task [37]. They are well suited for tasks where the number of object classes is perhaps endless, and classification is not applicable. The approach is to compute a certain distance metric between input samples and reference prototypes. Moreover, the training will not even require defective samples if the class features are well defined and distinct from each other. Unfortunately, textured objects present surface appearance and properties that are stochastic.

Different sampling strategies could be implemented to deal with *imbalanced datasets*. When the minority class represents the defective pieces, it could be convenient to use as many elements from the majority class as the available defective ones. This approach is known as *undersampling* in literature. It can certainly be applied when the amount of defective samples is adequate for the training task. The alternative for not reducing the majority class is to give the network the available defective samples multiple times, trying to get the same amount of the good ones. Such an approach is called *oversampling* in literature. It is important to notice that this method could be risky as it can be easy to overfit the network due to the scarce representation of the input space that usually cannot completely cover the possible scenarios. Nevertheless, there are cases in which the beforementioned solutions are valuable tools to enhance the performance of the classifier as the work proposed by Yap et al. [38].

An alternative approach that is often used to increase the robustness of the classification is *data augmentation*. Traditional techniques involve operations on the input

images, such as scaling, cropping, rotation, mirroring, and color shift [39]. The samples are augmented based on the available data with the risk of a strong correlation between the original samples and the augmented ones that could probably lead to overfitting scenarios on a small dataset. However, if the augmentation is correctly managed, it can boost the performance of the classifier. Indeed, data augmentation has been employed with success in many defect inspection methods [40, 41].

Other ways for enlarging the dataset have been experimented like passing the input data through an encoder-decoder network that applies different transformations featured with random noise [42]. Another approach worth mentioning is the generation of *virtual samples*. For example, the work presented by Leng et al. [43] successfully exploits virtual samples for face reconstruction.

Virtual data generation could be obtained by producing synthetic images with the intent to cover the whole input feature space. Generative adversarial network (GAN) [44] or the most recent conditional GAN (cGAN) [45] could be alternatively used for this purpose. However, this is computationally expensive and requires taking into account all possible configurations and boundary conditions for generating samples as close as possible to real ones. Domain randomization techniques [46] could be applied to synthetically generated data for improving the generalization capabilities and the robustness of the network.

Similar to humans, when learning new concepts or rules, if not clearly defined, the training can lead to fuzzy assumptions, possibly resulting in wrong outcomes. Additionally, when dealing with data obtained by a sensing apparatus, it is important to check the correctness of the acquired data samples to avoid possible causes of classification errors. A cleaning process should remove outliers (wrong data association of a sample with a class) and spurious samples that could confuse the learning process. Industrial processes often rely on qualitative evaluation, and unfortunately, different quality experts in the same industrial process classify the same product as belonging to different classes. If the same confusion is transferred to the DL architecture, the learning process will probably worsen the decision process. For this reason, a preprocessing stage on the data is essential. In most cases, the help of professionals of the sector for interpreting, filtering, and preprocessing the data is welcome.

A last and quite important aspect is the adoption of correct performance metrics and *loss functions* enabling successful training with *imbalanced datasets*. In this context, Mower [47] proposes a balanced accuracy statistic that mediates the *recall* and *specificity* metrics. A more general approach is to directly scale the confusion matrix terms based on the relative support of each class as proposed by Tripicchio et al. [48]. Other studies modify the *loss function* to account for class imbalance. In particular, *binary cross-entropy loss* is a common choice for classification tasks. In a study by Xi and Tu [49] a balanced *cross-entropy* is introduced where, differently from the *binary cross-entropy loss*, the contribution of the dominant class is multiplied by the fraction of the less dominant class. However, the method does not differentiate between easy/hard examples. A different approach is proposed by Lin et al. [50] where the authors focus the training on hard negatives, down-weighting the *loss* assigned to well-classified examples. The resulting *loss* is called *focal loss*.

5. Transfer learning for defect detection

It has been seen that the first layers of CNNs learn kernels acting as color blob detectors or Gabor filters. Such a property seems to be very general and the features

learned do not appear to be strictly dependent on the particular training set that has been adopted. As humans can learn from experience and transfer the notion learned in diverse application domains, similarly, a DL architecture can transfer the features learned on a particular dataset to another CNN, which will be trained on a different one [51]. Such a technique is called *transfer learning* and is a worthy tool for solving the problem of scarcity of defective samples. This paradigm involves pretraining the network on a dataset (usually larger) for learning feature extraction layers and afterward fine-tuning the classification pipeline with the relevant dataset for the specific task. Knowledge transfer breaks the fundamental assumption that the data presented to the network during the training phase must be in the same feature space as the ones presented in the inference phase. Feature extraction layers obtained applying the *transfer learning* paradigm would be able to extract generic convolution features that could be exploited in different tasks.

Following the *transfer learning* approach, the work by Sassi et al. [36] yielded a 97% *accuracy* during testing in the laboratory and proved successful during operation in a real production line, reaching an *accuracy* of 99% after subsequent training.

The work combines a traditional computer vision pipeline together with a DL architecture. This pipeline was necessary to maintain the compatibility with classical production lines and provide a correct input to the welding defect detection phase. The algorithm receives the raw image as input, converts it from Bayer format to grayscale, and improves the edge detection by equalizing the levels and applying a Gaussian blur. In a successive step, since different kinds of injectors can be analyzed by the same system, the type of injector is identified, and the position of its center is obtained. The algorithm proceeds to detect the outer shell of the injector head by estimating an external radius that approximates the detected blob. Then, using the extracted information, the algorithm performs an area search for welding points and estimates a welding circle on the joint. Subsequently, the algorithm collects statistics about the number of welding points found and their positions. In traditional industrial systems, a set of thresholds decided by the manufacturing company is used to evaluate the welding quality from the measured quantities.

A schematic overview of the algorithm is shown in **Figure 4**. The algorithm's output gives quantitative information about the welding and produces a processed image to be given as input to the second analysis stage. The extracted information allows evaluating the continuity of the welding in a certain area on the injector's head, verifying the centering of the inner part of the injector with respect to the outer one,

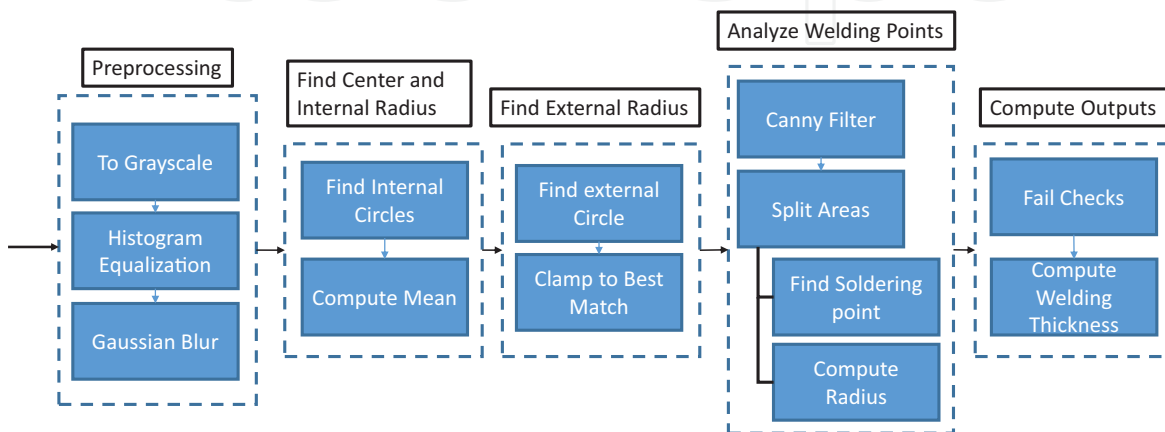


Figure 4. Schematics of the components of the geometrical analysis pipeline.



Figure 5.
 Schematic representation of the layers and blocks in the DenseNet-121 deep learning architecture.

and eventually the welding thickness. This information is also beneficial to clean the image from unnecessary data for the subsequent analysis and to center the injector images to obtain more controlled conditions on the input of the successive stage.

The DL architecture chosen in that work is the DenseNet-121. **Figure 5** depicts the structure of the network. DenseNet efficiently simplifies the connectivity pattern between layers guaranteeing maximum information flow by reusing the features through the network. Concerning the training phase, every layer has direct access to the gradients from the original input image and the *loss function*. Different from the first feedforward neural networks that connect the output of each layer to the subsequent layer after applying a composite of operations, DenseNet concatenates the output feature maps of the layers to obtain the equation $x_l = H_l([x_0, x_1, \dots, x_{l-1}])$. The network is formed by dense blocks, which have a constant size of the feature maps within a block but a varying number of filters, and transition layers that connect the blocks combining batch normalization, 1×1 convolution, and 2×2 pooling.

In the approach presented by Sassi et al. [36], the *transfer learning* technique has been employed and evaluated by comparing the results achieved when the features are transferred from a pretrained model on the *Material in Context* (MINC) [52] dataset. Such dataset contains 2,996,674 patches obtained from 436,749 images labeled according to 23 material classes. A binary classification problem has been set up by selecting positive samples as scrap injectors and negative samples as good injectors. The results of the classification problem are shown as a confusion matrix having four possible values—*true negative* (tn), *true positive* (tp), *false negative* (fn), and *false positive* (fp). The metrics that better estimate the quality of the defect analysis are the *recall* $\left(\frac{tp}{tp+fn}\right)$, which describes the ability to detect faulty pieces, and the *accuracy* $\left(\frac{tp+tn}{tp+tn+fp+fn}\right)$ that describes the overall quality of the analysis. The *precision* $\left(\frac{tp}{tp+fp}\right)$ is important to not discard too many injectors, but it is not crucial as the *recall* since not detecting a defect could be dangerous if it proceeds through the assembly line.

Unfortunately, the MINC dataset is highly unbalanced. Therefore, three classes, that is, *plastic*, *metal*, and *others*, have been selected as a subset for the pretraining stage to alleviate training problems. The idea of using such a dataset for *transfer learning* was to exploit the metallic features that could resemble the ones in the welding images, and the dataset class reduction does not affect the learned features on metallic materials.

6. Managing production variability

Sometimes, during production lines maintenance or innovations, the replacement of a machine, the change of a supplier, or the change in a manufacturing process, could lead to a significant variation on the usual production procedure in terms of the

visual quality of the products. Such situations could vanish the capacity of a machine computation to return the expected results.

In this context, continuing on the problem of detecting welding defects on injectors heads, the work presented by Tripicchio et al. [48] proposes possible solutions to this issue without requiring an architectural change in the learning architecture. The new case had to handle some modifications concerning the parameters associated with the welding process, producing input samples with specific artifacts that the previously designed and trained network did never encounter. In particular, such new inputs were correlated to a variation in the substance used for the soldering that generated gold-violet spots on the injector head in random positions. Such noise introduces a novel complexity in the detection of the defects because the spots can hide or visually resemble the presence of bumps and holes in the welding layer. The followed approach was to make fewer changes as possible in the architecture of the network, operating a smart preprocessing and applying filtering techniques.

The results show the ability to train a network with almost 7 million parameters on just 306 training images belonging to the new alteration, achieving a *recall* of 100.00% and an *accuracy* of 97.22%.

Such a result has been achieved leveraging on two important aspects. The first is the design of a custom preprocessing and filtering stage, while the second is the adoption of a novel data balancing strategy.

A preprocessing stage is needed on the input images with the aim of erasing or smoothing the chromatic nuances that could confuse the feature learning process. In particular, three filtering approaches have been proposed and tested (**Figure 6**). The first filter (*constant filter*) detects regions on the image in the gold and violet ranges of the HSV space filling such regions with a constant RGB color resembling the chromatic value of the injector contour. The second kind of filter (*median filter*), once selected the regions, fills them with the median RGB value of each channel. In the third filtering approach (*patch filter*), a 4×4 patch is virtually generated to resemble a part not affected by defects, and it is used to fill the detected gold and violet regions. In particular, every pixel of the regions is substituted with the value of the corresponding pixel of the synthetic image, adding the median value of the original image.

Different analyses have been done to assess the performance improvement given by such filters. As a result, a *patch filter* was selected as the method for the subsequent tuning of the network.

Concerning data imbalance, an exploration of different unbalanced splits has been performed. To prevent overfitting and lead the learning process toward generalization, the authors propose to compute the performance metrics at each evaluation step considering the input imbalance. In particular, metrics like *specificity* or *recall* are not affected by the imbalance of the data different from other metrics, like *accuracy*, which should be revised.



Figure 6. Different filters applied on a sector of the same injector contour image. (a) No filter. (b) Median fill filter. (c) Patch filter.

Defective injectors were chosen as positive samples and *false negatives* and *true positives* values were weighted depending on the imbalance since the defective class was the smaller one. The imbalance is compensated by multiplying these values by the proportion of the input dataset. Consequently, the confusion matrix presents a more balanced indication of the performance of the training.

Cross-validation has been applied to improve generalization concerning the stochastic gradient descent optimization. The network has been trained multiple times by combining different variations of the proportions between defective and good samples and changing the numbers of epochs. During the training phase, each epoch is compared with all previous epochs for obtaining the one with the highest performance in terms of *recall* and *accuracy*.

The *F-score* was chosen as the best multi-performance metric to evaluate the testing achieved on the different variations of the training. Concerning the imbalance, the obtained performances give that an *unbalanced dataset* could provide better results if the imbalance is considered while training.

7. Conclusions

This chapter highlights the importance of the employment of deep learning architectures in the context of future industrial applications with a focus on welding and welding defects detection. The industrial sector and especially the manufacturing industry pose several challenges to the design of efficient and robust quality inspection processes. The most common issues are discussed in detail, and possible countermeasures are suggested to overcome such issues. In particular, the problem of data imbalance, scarcity of examples, environmental noises, change in the nominal conditions of the process, or the presence of artifacts are discussed. Application examples from previous works of the authors are proposed to clarify how the suggested countermeasures can be put into practice. Although many industries are still scared of adopting deep learning approaches due to a lack of knowledge of their internal processes or reasoning, extensive use of artificial intelligence applications is envisaged for the near future.

Abbreviations

AI	Artificial Intelligence
CNN	Convolutional Neural Network
DL	Deep Learning
DNN	Deep Neural Network
GAN	Generative Adversarial Network
ICT	Information and Communication Technologies
IoT	Internet of Things
PLC	Programmable Logic Controller
ROI	Region of Interest
RSW	Resistance Spot Welding

IntechOpen


IntechOpen

Author details

Paolo Tripicchio* and Salvatore D'Avella
Department of Excellence in Robotics and AI, Institute of Mechanical Intelligence,
Scuola Superiore Sant'Anna, Pisa, Italy

*Address all correspondence to: paolo.tripicchio@santannapisa.it

IntechOpen

© 2022 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] D'Avella S, Tripicchio P. Supervised stowing as enabling technology for the integration of impaired operators in the industry. *Procedia Manufacturing*. 2020; **51**:171-178
- [2] Fraga-Lamas P, Fernández-Caramés TM, Blanco-Novoa ÓS, Vilar-Montesinos MA. A review on industrial augmented reality systems for the industry 4.0 shipyard. *IEEE Access*. 2018; **6**:13358-13375
- [3] Tofail SAM, Koumoulos EP, Bandyopadhyay A, Bose S, O'Donoghue L, Charitidis C. Additive manufacturing: Scientific and technological challenges, market uptake and opportunities. *Materials Today*. 2018; **21**(1):22-37
- [4] Manavalan E, Jayakrishna K. A review of internet of things (IoT) embedded sustainable supply chain for industry 4.0 requirements. *Computers & Industrial Engineering*. 2019; **127**:925-953
- [5] Tripicchio P, Unetti M, D'Avella S, Buffi A, Motroni A, Bernardini F, et al. A synthetic aperture uhf rfid localization method by phase unwrapping and hyperbolic intersection. *IEEE Transactions on Automation Science and Engineering*. 2021. DOI: 10.1109/TASE.2021.3057433
- [6] Tripicchio P, D'Avella S. Is deep learning ready to satisfy industry needs? *Procedia Manufacturing*. 2020; **51**: 1192-1199
- [7] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems – Volume 1, NIPS'12*. USA: Curran Associates Inc.; 2012. pp. 1097-1105
- [8] Malamas EN, Petrakis EGM, Zervakis M, Petit L, Legat J-D. A survey on industrial vision systems, applications and tools. *Image and Vision Computing*. 2003; **21**(2):171-188
- [9] Shao J, Yan Y. Review of techniques for on-line monitoring and inspection of laser welding. *Journal of Physics: Conference Series*. 2005; **15**(1): 101-107
- [10] Zheng K, Li J, Lei Tu C, Song Wang X. Two opposite sides synchronous tracking x-ray based robotic system for welding inspection. In: *2016 23rd International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*. Piscataway, NJ: IEEE; 2016. pp. 1-5
- [11] Zhao J, Sheng H, Zhou X. Study on the application of acoustic emission testing technique in monitoring 16 mn steel welding defects. In: *2016 International Conference on Advanced Mechatronic Systems (ICAMechS)*. Piscataway, NJ: IEEE; 2016. pp. 318-321
- [12] Fang Z, Xu D, Tan M. A vision-based self-tuning fuzzy controller for fillet weld seam tracking. *IEEE/ASME Transactions on Mechatronics*. 2011; **16**(3):540-550
- [13] Podržaj P, Čebular A. The application of lvq neural network for weld strength evaluation of RF-welded plastic materials. *IEEE/ASME Transactions on Mechatronics*. 2016; **21**(2):1063-1071
- [14] Senthil Kumar G, Natarajan U, Ananthan SS. Vision inspection system for the identification and classification of defects in MIG welding joints. *International Journal of Advanced*

Manufacturing Technology. 2012;**61**
(9–12):923-933

[15] Zahran O, Kasban H, El-Kordy M, Abd El-Samie FE. Automatic weld defect identification from radiographic images. *NDT & E International*. 2013;**57**:26-35

[16] Yan Z, Zhang G, Wun L. Simulation and controlling for weld shape process in p-gmaw based on fuzzy logic. In: 2011 IEEE International Conference on Mechatronics and Automation. Piscataway, NJ: IEEE; 2011. pp. 2078-2082

[17] Zammar IA, Mantegh I, Huq MS, Yousefpour A, Ahmadi M. Intelligent thermal control of resistance welding of fiberglass laminates for automated manufacturing. *IEEE/ASME Transactions on Mechatronics*. 2015; **20**(3):1069-1078

[18] Masci J, Meier U, Ciresan D, Schmidhuber J, Fricout G. Steel defect classification with Max-Pooling Convolutional Neural Networks. In: 2012 Int. Jt. Conf. Neural Networks. Piscataway, NJ: IEEE; 2012. pp. 1-6

[19] Zhou S, Chen Y, Zhang D, Xie J, Zhou Y. Classification of surface defects on steel sheet using convolutional neural networks. *Materials and Technology*. 2017;**51**:123-131

[20] Ronay Ak, Max Ferguson, Yung-Tsun T. Lee, and Kincho H. Law. Automatic localization of casting defects with convolutional neural networks. In: 2017 IEEE International Conference on Big Data (BigData 2017). Boston, MA: IEEE; 2017

[21] Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision

and Pattern Recognition. Piscataway, NJ: IEEE; 2014. pp. 580-587

[22] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*. 2015;**28**:91-99

[23] Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE; 2016. pp. 779-788

[24] Ruisz J, Biber J, Loipetsberger M. Quality evaluation in resistance spot welding by analysing the weld fingerprint on metal bands by computer vision. *The International Journal of Advanced Manufacturing Technology*. 2007;**33**(9):952-960

[25] Girshick R. Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE; 2015. pp. 1440-1448

[26] Redmon J, Farhadi A. Yolo9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE; 2017. pp. 7263-7271

[27] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Cheng-Yang F, et al. Ssd: Single shot multibox detector. In: European Conference on Computer Vision. New York, NY: Springer; 2016. pp. 21-37

[28] Redmon J, Farhadi A. Yolov3: An incremental improvement. In: Computer Vision and Pattern Recognition (CVPR). Berlin/Heidelberg, Germany: Springer; 2018. pp. 1804-2767. arXiv preprint arXiv:1804.02767

- [29] Lin T-Y, Dollár P, Girshick R, He K, Hariharan B, Belongie S, editors. Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE; 2017. pp. 2117-2125
- [30] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: The 3rd International Conference on Learning Representations. San Diego, CA: ICLR; 2015. arXiv preprint arXiv:1409.1556
- [31] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE; 2016. pp. 770-778
- [32] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE; 2017. pp. 4700-4708
- [33] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. 2017. arXiv preprint arXiv:1704.04861
- [34] Dai W, Li D, Tang D, Jiang Q, Wang D, Wang H, et al. Deep learning assisted vision inspection of resistance spot welds. *Journal of Manufacturing Processes*. 2021;62:262-274
- [35] Howard A, Sandler M, Chu G, Chen LC, Chen B, Tan M, et al., editors. Searching for mobilenetv3. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE; 2019. pp. 1314-1324
- [36] Sassi P, Tripicchio P, Avizzano CA. A smart monitoring system for automatic welding defect detection. *IEEE Transactions on Industrial Electronics*. 2019;66(12):9641-9650
- [37] Ren R, Hung T, Tan KC. A generic deep-learning-based approach for automated surface inspection. *IEEE Transactions on Cybernetics*. 2018; 48(3):929-940
- [38] Yap BW, Rani KA, Rahman HAA, Fong S, Khairudin Z, Abdullah NN. An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets. In: Herawan T, Deris MM, Abawajy J, editors. Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013). Singapore: Springer Singapore; 2014. pp. 13-22
- [39] Cubuk ED, Zoph B, Mané D, Vasudevan V, Le QV. Autoaugment: Learning augmentation policies from data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE; 2019. pp. 113-123. ArXiv, abs/1805.09501
- [40] Wang T, Chen Y, Qiao M, Snoussi H. A fast and robust convolutional neural network-based defect detection model in product quality control. *The International Journal of Advanced Manufacturing Technology*. 2018;94(9):3465-3471
- [41] Chen J, Liu Z, Wang H, Núñez A, Han Z. Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network. *IEEE Transactions on Instrumentation and Measurement*. 2018;67(2):257-269
- [42] Jorge J, Vieco J, Paredes R, Sánchez J-A, Benedí J-M. Empirical evaluation of

variational autoencoders for data augmentation. In: VISIGRAPP. Setúbal, Portugal: SCITEPRESS; 2018

[43] Leng B, Yu K, Jingyan QIN. Data augmentation for unbalanced face recognition training sets. *Neurocomputing*. 2017;235(C):10-14

[44] Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial nets. In: NIPS. Red Hook, NY: Curran Associates, Inc.; 2014

[45] Mirza M, Osindero S. Conditional generative adversarial nets. 2014. cite arxiv:1411.1784

[46] Tobin J, Fong R, Ray A, Schneider J, Zaremba W, Abbeel P, editors. Domain randomization for transferring deep neural networks from simulation to the real world. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway, NJ: IEEE; 2017. pp. 23-30

[47] Mower JP. Prep-Mt: Predictive RNA editor for plant mitochondrial genes. *BMC Bioinformatics*. 2005; 6(1):96

[48] Tripicchio P, Camacho-Gonzalez G, D'Avella S. Welding defect detection: Coping with artifacts in the production line. *The International Journal of Advanced Manufacturing Technology*. 2020;111(5):1659-1669

[49] Xie S, Zhuowen T. Holistically-nested edge detection. In: Proceedings of the IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE. 2015; pp. 1395-1403

[50] Lin TY, Goyal P, Girshick R, He K, Dollár P. Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer

Vision. Piscataway, NJ: IEEE. 2017. pp. 2980-2988

[51] Yosinski J, Clune J, Bengio Y, Lipson H. How transferable are features in deep neural networks? In: NeurIPS. Red Hook, NY: Curran Associates, Inc.; 2014

[52] Bell S, Upchurch P, Snavely N, Bala K. Material Recognition in the Wild with the Materials in Context Database. Piscataway, NJ: IEEE; 2014