UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E INFORMÁTICA INDUSTRIAL

SYLVIO ALEXANDRE BIASUZ BLOCK

DETECÇÃO E CLASSIFICAÇÃO AUTOMÁTICA DE DEFEITOS EM SUPERFÍCIES METÁLICAS UTILIZANDO TÉCNICAS DE APRENDIZADO PROFUNDO

TESE

CURITIBA
2024

SYLVIO ALEXANDRE BIASUZ BLOCK

DETECÇÃO E CLASSIFICAÇÃO AUTOMÁTICA DE DEFEITOS EM SUPERFÍCIES METÁLICAS UTILIZANDO TÉCNICAS DE APRENDIZADO PROFUNDO

Automatic detection and classification of defects in metallic surfaces by using deep learning techniques

Tese apresentado(a) como requisito para obtenção do título(grau) de Doutor em Engenharia Elétrica e Informática Industrial, do Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, da Universidade Tecnológica Federal do Paraná (UTFPR).

Orientador: Prof. Dr. Rodrigo Minetto
Coorientador: Prof. Dr. Ricardo Dutra da Silva

CURITIBA

2024

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Curitiba

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

SYLVIO ALEXANDRE BIASUZ BLOCK

DETECÇÃO E CLASSIFICAÇÃO AUTOMÁTICA DE DEFEITOS EM SUPERFÍCIES METÁLICAS UTILIZANDO TÉCNICAS DE APRENDIZADO PROFUNDO

Trabalho de pesquisa de doutorado apresentado como requisito para obtenção do título de Doutor Em Ciências da Universidade Tecnológica Federal do Paraná (UTFPR). Área de concentração: Engenharia De Computação.

Data de aprovação: 16 de Outubro de 2024

Dr. Rodrigo Minetto, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Eduardo Todt, Doutorado - Universidade Federal do Paraná (Ufpr)

Dr. Helio Pedrini, Doutorado - Universidade Estadual de Campinas (Unicamp)

Dr. Joao Alberto Fabro, Doutorado - Universidade Tecnológica Federal do Paraná

Dra. Myriam Regattieri De Biase Da Silva Delgado, Doutorado - Universidade Tecnológica Federal do Paraná

RESUMO

BLOCK, Sylvio Alexandre Biasuz. **Detecção e classificação automática de defeitos em superfícies metálicas utilizando técnicas de aprendizado profundo**. 2024. 66 f. Tese (Doutorado em Engenharia Elétrica e Informática Industrial) – Universidade Tecnológica Federal do Paraná. Curitiba, 2024.

Esta tese investiga a detecção e classificação automáticas de defeitos superficiais em peças metálicas durante as etapas de estampagem e soldagem na indústria automotiva. Na primeira parte desta pesquisa, o foco foi a detecção de defeitos de impressão durante a fase de estampagem. Esses defeitos, que se assemelham a picos, são tipicamente causados pela presença de sujeira ou detritos na maquinaria de prensagem. Dependendo da natureza e extensão do defeito, a peça veicular afetada pode precisar ser reparada ou descartada. Uma rede neural convolucional, conhecida como RetinaNet, foi utilizada para detectar e classificar os defeitos de impressão em categorias leve e severa. Além disso, foi explorada a coerência temporal, rastreando regiões detectadas em quadros consecutivos de um vídeo, a fim de reduzir alarmes falsos — regiões candidatas instáveis que raramente são (re)detectadas — ou para corrigir a classificação de algumas regiões que são classificadas alternadamente entre as duas categorias de defeitos. Nos experimentos, foi alcançada uma precisão média (mAP) de 76.21% para detectar e classificar defeitos de impressão leves e severos. No entanto, ao considerar apenas as impressões severas, os valores de precisão e revocação alcançaram 90% e 92%, respectivamente. Esses resultados são promissores, pois defeitos leves não são críticos e podem ser corrigidos ao longo da linha de produção. Para a etapa de soldagem, variações da arquitetura de aprendizado profundo YOLOv7 foram exploradas para detectar e classificar quatro tipos comuns de defeitos — poros, depósitos, descontinuidades e manchas. Foi desenvolvida uma rede leve, adequada para dispositivos de borda industriais, que alcançou um mAP de 69% para classificação detalhada de defeitos e 77% para uma classificação ampla. O estudo também introduz o LoHi-WELD, um novo conjunto de dados público que contém 3.022 imagens anotadas de cordões de solda, obtidas a partir de um processo de soldagem robótica Metal Active Gas (MAG), abrangendo tanto imagens de baixa quanto de alta resolução. Ambas as pesquisas foram realizadas em colaboração com um parceiro da indústria, que forneceu imagens das peças veiculares, bem como informações cruciais sobre os tipos de defeitos, frequência de ocorrência e requisitos de qualidade. Os resultados evidenciam o potencial para a implementação de soluções robustas para inspeção de defeitos em ambientes industriais.

Palavras-chave: Defeitos em estampagem. Defeitos em soldagem. Aprendizagem profunda. Detecção. Classificação.

ABSTRACT

BLOCK, Sylvio Alexandre Biasuz. **Automatic detection and classification of defects in metallic surfaces by using deep learning techniques**. 2024. 66 p. Thesis (PhD in Electrical Engineering and Industrial Informatics) – Universidade Tecnológica Federal do Paraná. Curitiba, 2024.

This thesis investigates the automatic detection and classification of surface defects on metal parts during the stamping and welding processes within the automotive industry. In the first part of this research, the focus was on detecting imprint defects during the stamping stage. These defects, which resemble peaks, are typically caused by the presence of dirt or debris in the press machinery. Depending on the nature and extent of the defect, the affected vehicle part may need to be repaired or discarded. A convolutional neural network, RetinaNet, was used to detect and classify imprint defects into the categories mild and severe. Additionally, temporal coherence was explored by tracking detected regions across consecutive frames of a video to reduce false alarms — unstable candidate regions that are rarely (re)detected many times — and to correct the classification of some regions that are alternated classified into the two classes of defects. In the experiments, a mean average precision (mAP) of 76.21% was achieved for detecting and classifying mild and severe imprint defects. However, when considering only severe imprints, precision and recall values reached 90% and 92%, respectively. These are promising results, as mild defects are not critical and can be fixed along the production line. For the welding part, variations of the baseline YOLOv7 deep learning architecture were explored to detect and classify four common types of defects — pores, deposits, discontinuities, and stains. A lightweight network, suitable for industrial edge devices, was developed, achieving a mAP of 69% for fine-grained defect classification and 77% for coarse classification. The study also introduces LoHi-WELD, a novel public dataset comprising 3,022 annotated images of weld beads, from a Metal Active Gas (MAG) robotic welding process, encompassing both low and high-resolution imagery. Both researches had the collaboration of an industry partner, who provided image data of the vehicle parts as well as crucial information about defect types, their frequency, and quality requirements. The research findings highlight the potential for implementing robust defect detection solutions in industrial settings.

Keywords: Stamping defects. Welding defects. Deep learning. Detection. Classification.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

In this chapter we present the problem of automated inspection techniques for detecting defects on metallic surfaces in the automotive industry's stamping and robotic welding processes. Thus emphasizes the critical role of automation in ensuring product quality and provides detailed explanations of the vehicle manufacturing process, including the stamping and welding stages. By highlighting the specific challenges in these processes, the chapter illustrates how the proposed techniques enhance defect detection and classification. Finally, it summarizes the contributions of this work to advancing automated inspection technologies and improving quality assurance practices in automotive manufacturing.

1.1 MOTIVATION

Automated inspection in industry is essential for ensuring the quality of many processes before, during and after the production lines. This is why research in this field is very active, with many proposed methods and an extensive bibliography (SUN *et al.*, 2019; WANG *et al.*, 2019; ZOU *et al.*, 2019; CHANG; PARK, 2019; SUN *et al.*, 2017). This kind of automation has been applied to a wide range of problems, including fabric defect detection, as reviewed by (HANBAY *et al.*, 2016), and defect detection in printed circuit boards (ANITHA; RAO, 2017). The inspection of metallic surfaces stands out as a critical application, particularly in the automotive industry, where the quality of metal parts is directly tied to vehicle quality and safety (SUN *et al.*, 2018).

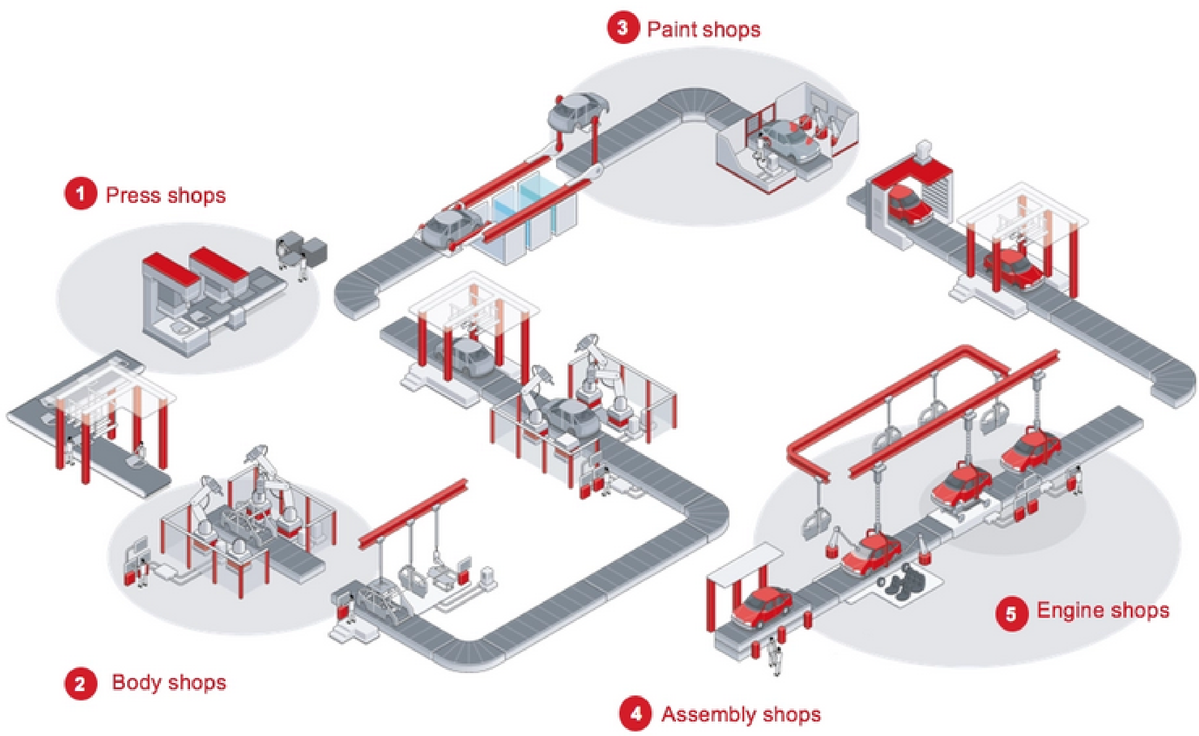
Surface inspection of metal parts is, therefore, a crucial task to ensure the quality of vehicles produced in the automotive industry. The primary objective of this inspection is to detect anomalies and defects in vehicle parts, preventing problems that originate early in the production process from being carried through the production chain, which can lead to a loss of product quality and material waste.

In this context, computer vision systems that use digital images for product quality inspection are being tested for fast detection and classification of defects in metallic parts (MALAMAS *et al.*, 2003). However, it is worth nothing, that many quality control and management tasks are still performed by specialized operators. These tasks are susceptible to human fatigue, distraction, or even subjective interpretation, which can vary from one operator to another. Automated systems offer a robust, reliable, and repeatable alternative for inspection along an assembly line.

1.2 RESEARCH PROBLEMS

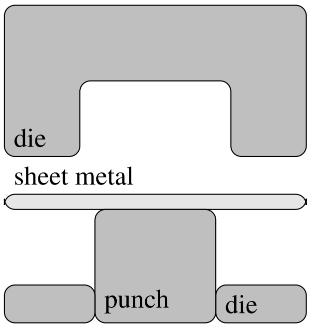
This thesis presents an in-depth investigation into the automatic inspection of defects in metallic surfaces within two critical stages of the automotive production chain: the stamping and robotic welding processes. As illustrated in Figure 1, the vehicle manufacturing process begins in the press (or stamping) shop, where vehicle parts are molded. These parts are then transported to the body shop, where the assembly of the vehicle starts through the welding process. After welding, the vehicle undergoes painting and is subsequently moved to the assembly and engine shops, where the power train is installed and other vehicle components are assembled.

Figure 1 – Production line process of an automotive industry. Source: Beck & Pollitzer industrial installation and machine relocation services.

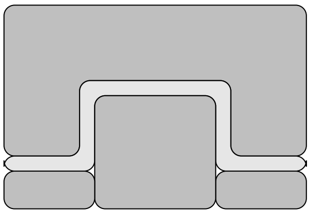


In the automotive industry, stamping is the initial stage of vehicle production and the main one with regard to the conformation of metallic material. This process transforms flat metal sheets into shapes to build a vehicle. Figure 2 illustrates a basic stamping machinery system and its essential components used to produce a deformed automotive part (Figure 2 (a)). Essentially, a punching force is applied to the metal sheet, while a die shapes the surface (Figure 2 (b)).

Figure 2 – Stamping machinery: a punching force deforms a sheet metal using a modeling die. Source: Author.



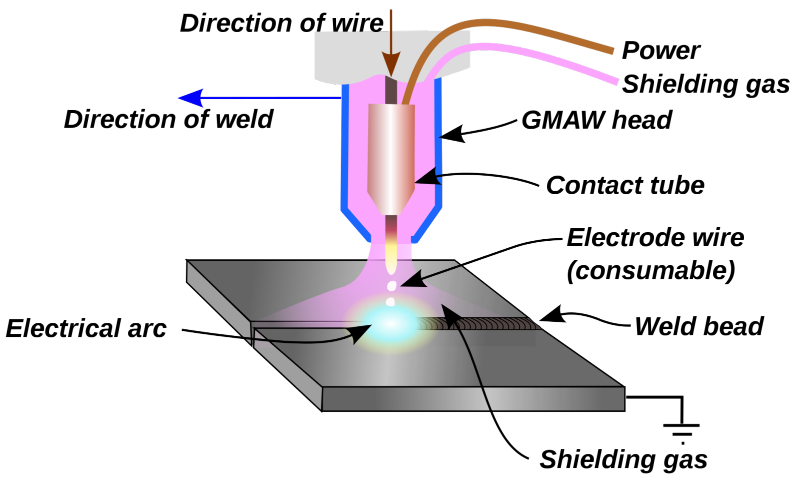
(a) Press machinery.



(b) Deformed sheet metal.

After stamping, the next stage in vehicle construction is the robotic welding of parts to form the vehicle's structure. In this process, a weld bead is created by depositing filler material into the joint between two metal pieces, as illustrated in Figure 3. This technique is used when the joining of two or more parts requires great structural rigidity, particularly in areas critical to the vehicle's safety.

Figure 3 – Illustration of the Metal Active Gas (MAG) welding process, highlighting the interaction between the welding arc, electrode, shielding gas, and base material during the formation of a weld joint. Source: Wikimedia Commons.



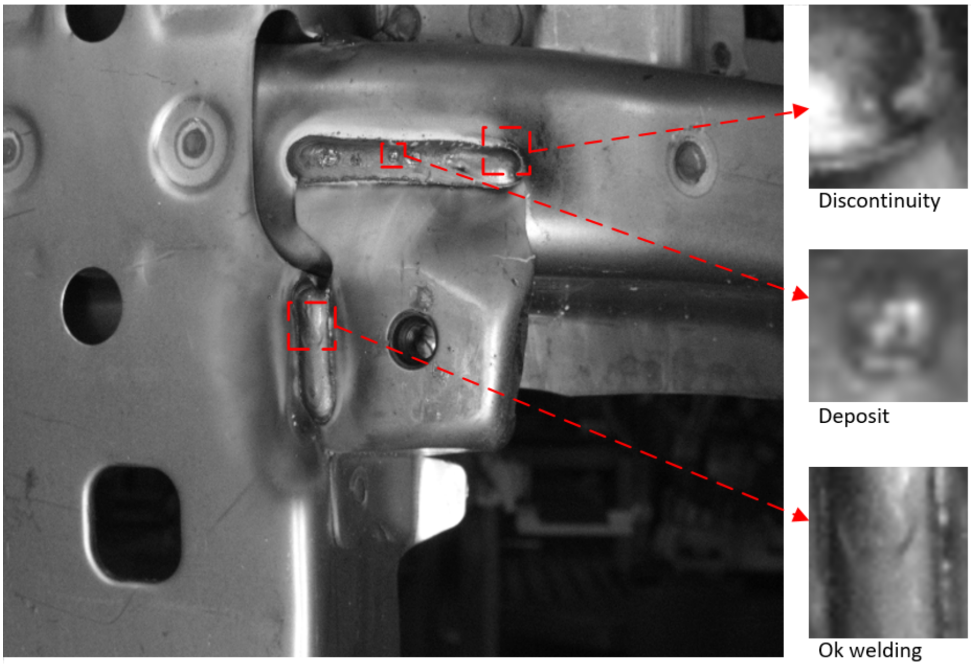
Regarding the stamping process, this thesis focuses on the automatic detection and classification of a defect known as imprint, which can occur on the surface of the metal parts. Imprints are peak-like surface deformations caused by machinery pressing against metal sheets contaminated with dirt or debris. Figure 4 illustrates examples of such defects. Imprint is the most common imperfection that may occur in the production process of stamped metal parts, accounting for 61.84% of all defects, according to the international automotive industry partner in this research. If not quickly detected, the problem will incur in extra costs due to repainting, rescheduling of the production line, or the disposal of substantial quantities of irrecoverable surfaces.

Figure 4 – Samples of imprint defects (highlighted by red circles). Source: Author.

In the stamping process, imprint control is currently performed through visual inspection by human experts, a difficult, time-consuming, and costly task. For instance, since luminance can vary across stamped parts, the expert must consider multiple observation points to ensure a reliable assessment. Also, due to the non-stop production lines in the industry, only a small fraction of the samples is inspected — about one out of every 50, even for exposed parts such as a hood.

In the robotic welding process, this study focuses on detecting and classifying four types of defects: discontinuity, deposit, pore, and stain. Similar to the stamping process, welding requires visual inspection by qualified operators, but with the aggravating difficulty of accessing regions of "robotized islands". These islands are physically isolated by fences, where welding and assembly robots operate. The need for isolation exists due to the risk of collision between robots and humans. Accessing these areas requires turning off all robots within the island, making each entry costly for the production line. Beyond visual quality inspection, the physical integrity of welds is critical to vehicle safety, leading to specific regulations. Different safety standards define the percentage of parts that must be visually inspected, with some regulations requiring 100% inspection. Figure 5 shows two regions with different defects — deposit and discontinuity — alongside a weld region considered to be of good quality.

Figure 5 – Example of robotic welding beam: a discontinuity, a deposit defect, and a welding region of good quality are highlighted in red and zoomed in. Source: Author.



1.3 RESEARCH GOALS AND CONTRIBUTIONS

The goal of this research project is to develop robust techniques for the automated inspection of visual defects on metallic surfaces of the automotive industry, with a focus on enhancing both the accuracy and efficiency of defect detection and classification. This involves the creation of frameworks tailored to address the specific challenges encountered in stamping and welding processes. Additionally, the project provides publicly accessible and annotated image datasets to enable further research and facilitate the comparison of different methodologies. Moreover, this research aims to contribute to the advancement of automated inspection technologies and the improvement of quality assurance processes in the automotive manufacturing sector.

Specifically for the imprint stamping problem, we proposed a framework (BLOCK *et al.*, 2021) to automatically detect and classify defects into two categories: *mild imprint*, for which the expected surface quality can be obtained after polishing and painting; and *severe imprint*, for which the expected quality is unattainable, necessitating the disposal of the stamped part. We used for this part a convolutional neural network (CNN), known as RetinaNet (LIN *et al.*, 2017), to detect and classify the imprint defects; and a filter-based tracker, known as Minimum Output Sum of Squared Error (MOSSE) (BOLME *et al.*, 2010), to compute the trajectories of such defects during the inspection. The framework performance was evaluated over 31 videos spanning more than 14,000 image frames. In our experiments, we achieved a mean average precision (mAP) of 76.21% for detecting and classifying mild and severe defects. When considering only severe imprints, we obtained precision and recall values of 90% and 92%, respectively. These results are promising, especially as mild defects are not critical and can be fixed along the production line. Additionally, we show that the proposed framework outperforms other object detectors, even when combined in an ensemble.

In summary, the contributions for the imprint problem are: (1) the proposal of a system to assist the automotive industry by inspecting every single produced vehicle part; (2) a strategy that combines detection, classification, and tracking in a synergistic way, such that by tracing the trajectory of potential imprint defects, it is possible to monitor how many times a surface region was (re)detected, as well as the classification labels assigned to it — false alarms are usually unstable, alternating classifications into mild and severe, and are not detected consistently through the frames; and (3) a publicly available dataset with several image sequences of vehicle parts, acquired by a multi-camera image acquisition system from an international automotive industry, that hopefully will be useful to other researchers working on this problem — the dataset has associated ground-truth annotations obtained with the help of a quality control specialist.

For the welding problem, we investigate a framework to automatically detect and classify defects using a YOLOv7 (WANG *et al.*, 2023) convolutional neural network. The network detects and classifies real weld defects produced during the robotic welding process. We evaluated the efficiency of different YOLOv7 model sizes, performed cross-validation experiments with low and high resolution image samples, and investigated the impact of image resolution on the detection and classification quality.

Specifically, the main contributions for the welding problem can be summarized as follows: (1) a novel dataset, referred to here as LoHi-WELD, composed of 3,022 images of regions containing real weld beads and spanning more than 22,000 weld defects, one of the largest in terms of the number of defects to the best of our knowledge; the dataset is divided into low resolution images (`weld-low` dataset) and high resolution images (`weld-high` dataset), and includes manual annotations of four categories of defects for research purposes; (2) an experimental evaluation of a robust deep learning network, YOLOv7 (WANG *et al.*, 2023), used as a baseline architecture to explore the proposed dataset by comparing the results of a tiny architecture model (YOLOv7-tiny), developed to run on edge devices with limited computational resources, with those of a higher computational cost model (YOLOv7); moreover, we evaluate the impact of different parameters, such as image size and data augmentation for the considered deep networks; and, (3) we explored cross-dataset experiments – a model trained for low resolution weld images was used for inference in high resolution weld images (and vice versa) – and fusion, by combining the `weld-low` and `weld-high` datasets. These contributions emerge from a comprehensive review of NDI methods for weld defect inspection, that was organized into two branches of artificial intelligence techniques widely used for this purpose (handcrafted and deep learning methods), categorized by the sensor type used during the image acquisition, and supplemented by a dataset overview table, that indicates the weld defects addressed, dataset size, total number of defects, availability, etc.

1.4 PUBLICATIONS

This research has produced the following publications (BLOCK *et al.*, 2021) and (BLOCK *et al.*, 2024):

- S. B. Block, R. D. da Silva, L. B. Dorini and R. Minetto, "**Inspection of Imprint Defects in Stamped Metal Surfaces Using Deep Learning and Tracking**," in *IEEE Transactions on Industrial Electronics*, vol. 68, no. 5, pp. 4498-4507, 2021.
- S. B. Block, R. D. da Silva, A. E. Lazzaretti and R. Minetto, "**LoHi-WELD: a novel industrial dataset for weld defect detection and classification, a deep learning study, and future perspectives**", in *IEEE Access* journal, vol. 12, pp. 77442 - 77453, 2024.

Furthermore, to promote experiment reproducibility, foster collaboration, and stimulate innovation in automated inspection and quality assurance for metallic surfaces, we have created GitHub repositories for both projects:

- https://github.com/SylvioBlock/Imprints
- https://github.com/SylvioBlock/LoHi-Weld

These repositories provide open access to the deep learning architectures, trained models, necessary dependencies, and datasets. By sharing these resources openly, we aim to facilitate further research, benchmarking of techniques, and the continuous advancement of surface defect detection in the automotive industry.

1.5 DOCUMENT STRUCTURE

This document is structured as follows. In Chapter 2, we review the relevant literature on surface inspection and weld defects. Chapter 3 describes the methodology employed for detecting and classifying stamping defects. In Chapter 4, we provide details about the equipment used for image acquisition, the dataset generated, and the experimental analysis of the proposed approach for stamping defects. Chapter 5 introduces our LoHi-WELD dataset, which includes defect annotations in weld beads, along with our framework for defect detection and classification. In Chapter 6, we present the results for welding defect detection using both low- and high-resolution images, evaluate the performance of the proposed framework, and discuss factors influencing its effectiveness. Finally, in Chapter 7 we provide conclusions regarding the developed research.

2 RELATED WORK

Industrial systems make it possible to inspect surfaces of the most diverse origins, contemplating a wide range of defects. The detection and classification of defects is a challenging problem and usually requires an efficient calibration of different types of sensors such as cameras and lighting systems, as emphasized by studies (LEON; KAMMEL, 2006; MARTÍNEZ *et al.*, 2013), as well as, custom-made solutions to ensure that the algorithms learn the optimal discriminative features representing the data.

Many algorithms for this task usually consider hand-crafted features such as those from image binarization (WIN *et al.*, 2015), bag-of-words for template matching (KONG *et al.*, 2017), wavelet transforms (LI *et al.*, 2006), and fractals (YAZDCHI *et al.*, 2009). In recent years, the advances in machine learning achieved by Convolutional Neural Networks (CNN) (LECUN *et al.*, 2015) inspired the design of novel architectures and strategies for a wide range of industrial applications (LIN *et al.*, 2017; REDMON *et al.*, 2016). Inspection systems based on convolutional neural networks have been applied in various industrial contexts, including the identification of bolt looseness (WANG *et al.*, 2019), detection of concrete cracks (ZOU *et al.*, 2019), wire mismatching (CHANG; PARK, 2019), fault diagnosis in induction motors (SUN *et al.*, 2017), bearing fault recognition (LIU *et al.*, 2019), and inspection of solder joints (CAI *et al.*, 2018).

In Section 2.1, we review relevant studies that are more closely related to imprint defects. It is important to note that, to the best of our knowledge, imprint defects have not been extensively addressed in the existing literature. In contrast, Section 2.2 provides a comprehensive review of the literature on welding defects, a research topic that has been much more thoroughly explored and studied.

2.1 GENERAL SURFACE DEFECTS

In this section, we review research related to surface inspection, with a particular emphasis on metallic surfaces, as these are more closely related to the inspection of imprints in stamped parts

Herrero *et al.* (2018) proposed an acquisition system composed of two light sources for inspecting rail surfaces. They used a blue and a red light, placed before and after a line scan camera, and observed that flat surfaces reflect the same amount of both light sources, while surfaces with volumetric defects do not. By analyzing the differences between the red and blue channels of the image, and applying standard image processing techniques — including foreground extraction, segmentation, and feature extraction based on texture and morphology attributes — they developed an image descriptor which was used to train a basic multilayer perceptron network for region classification. Their system requires calibration and assumes uniform light intensity across the surface and consistent reflective properties of the rail. The system detected 46.9% of the existing defects.

In the same year, Chen and Jahanshahi (2018) proposed a deep learning framework to identify cracks in metallic components of nuclear power plants. An underwater camera, combined with a robotic arm scanner, was used to record low-resolution videos, which were then employed to extract image patches of cracks and non-cracks for training and validation. They developed a convolutional neural network with four convolutional layers for crack detection, a Naive Bayes classifier to reduce false alarms, and a data fusion scheme to compute the spatiotemporal coherence of cracks throughout the video. A drawback of their system is the requirement for non-crack samples during training, which necessitates a large amount of training data and may generate false alarms for non-crack regions not seen during training.

Song *et al.* (2019) proposed a method to detect scratches on metal component surfaces by using a convolutional neural network to segment/binarize the image. The segmentation architecture used is U-Net (RONNEBERGER *et al.*, 2015). This network has a contracting phase that uses convolution and max-pooling operations to reduce the spatial dimensions of the image and extract feature information; and an expanding phase that applies deconvolutions and concatenations to increase image resolution and generate a segmentation map. False alarms were reduced by thresholding the segments based on fixed geometric criteria. According to the authors, a drawback of the technique is that the segmentation often produces many non-scratch regions.

Manish *et al.* (2018) investigated the surface inspection of grinding specimens using the Canny edge detection and histogram analysis. According to the authors, under constant lighting conditions, analyzing the grayscale intensity distribution in a histogram is useful for understanding the surface characteristics of different grinding types. As they observed, the peak values in the grayscale distribution, which ranges from 0 to 255, fall between 150 and 255 for smooth surfaces, 100 and 170 for rough surfaces, and 20 and 50 for defective surfaces. The proposed technique appears to be highly specific to the described application, which requires refinement and testing for use in similar scenarios, making it less scalable and robust.

(WEN *et al.*, 2018), a machine vision system for bearing roller surface inspection was proposed. To ensure product quality, a multitask convolutional neural network was designed for defect detection. Defect features were extracted using a shared convolutional neural network, and then defects were classified and their positions calculated simultaneously. The bearing roller was evaluated based on the position, category, and area of the defects. The best precision achieved was 92.59%. However, the small database and class imbalance may limit the method's representativeness in similar applications.

The system described by (IGLESIAS *et al.*, 2018) aimed to detect and classify quality grading issues in slate. The main goal is to detect sulphides and golden flower-like staining. The inspection algorithms were adapted to the characteristics of the slate mine, which provided 70 slabs for this study. Although the inspection algorithms performed well in detecting surface irregularities, warping, false squaring, material defects, and flower-like staining (both white and golden), the detection of sulphides was not sufficiently robust. However, results were satisfactory for calibration slabs used to test the performance of the sulphide detection algorithm.

There are also some technologies based on a principle known as *phase deflectometry* (KNAUER *et al.*, 2004). Similar to how humans inspect an object by positioning it towards the light, such systems (KNAUER *et al.*, 2004) use a flexible robotic arm coupled with a structured light projector to cast sinusoidal striped patterns onto a metal surface, and a calibrated camera to record the reflected patterns. Slope variations of the surface caused by defects produce changes in intensity and brightness, leading to pattern distortion. Using phase-shift algorithms (Kammel; Puente Leon, 2008), it is possible to precisely detect and measure these distortions. The main drawbacks are the high costs associated — commercial systems require multiple robotic systems for a real production line (Micro-Epsilon, 2019) — and the need for rigorous camera calibration and fine-tuning.

2.2 WELD INSPECTION

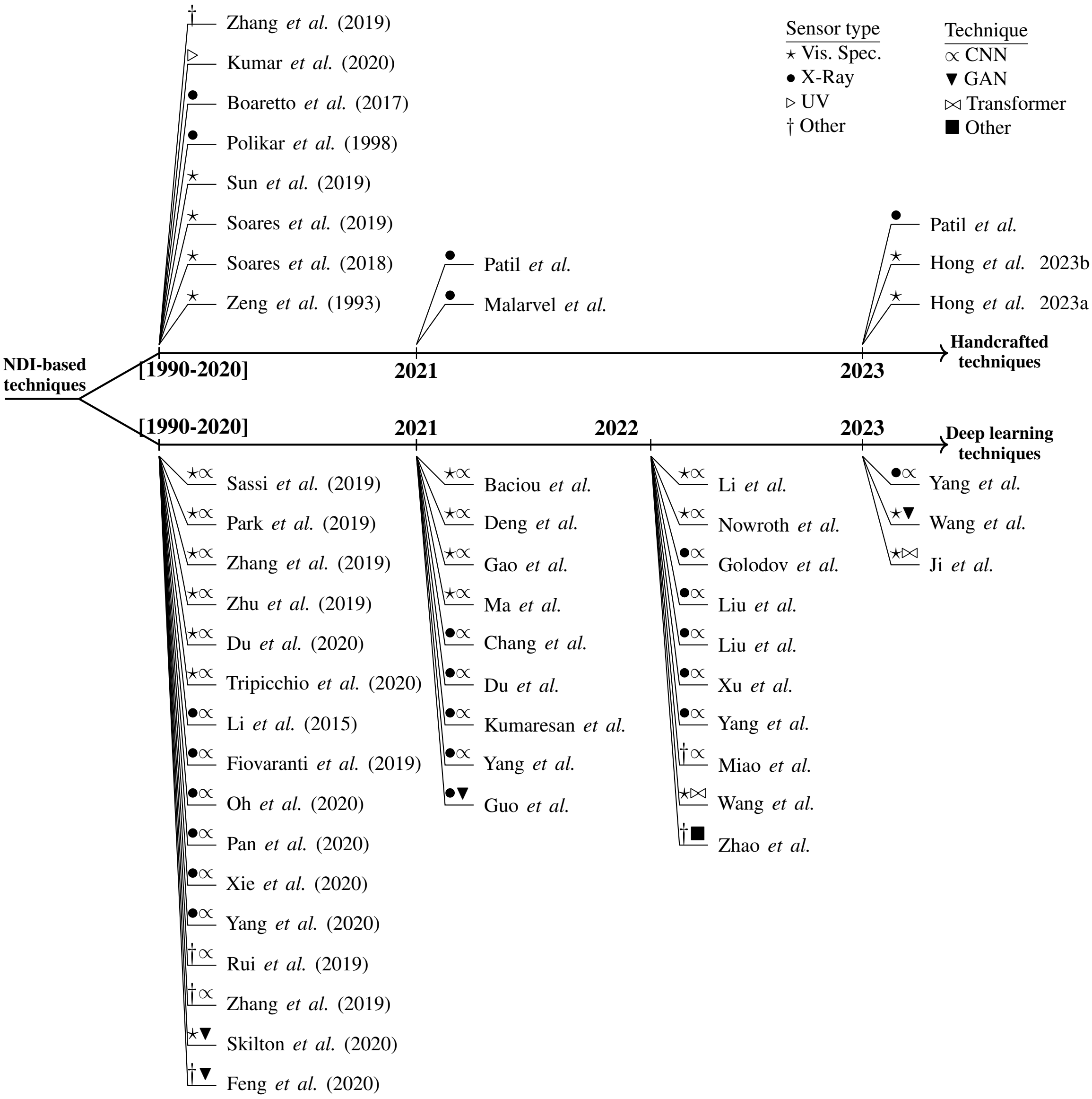
Welding inspection is a critical process that evaluates the quality of welded joints by looking for defects. It involves various techniques and methods, depending on the objectives of the inspection, the type of weld, and the material properties. There are two main categories of welding testing, those based on destructive methods and those based on non-destructive methods.

Destructive inspection involves removing a sample of the welded joint and subjecting it to various tests to evaluate its quality. Tensile testing, bend testing, and Charpy impact testing are the most common destructive tests. Tensile testing measures the strength of the welded joint by applying an axial load until failure. Bend testing evaluates the welded joint ductility by bending a sample to a specified angle. Charpy impact testing assesses the welded joint toughness by striking a notched sample with a pendulum and measuring the energy absorbed during the fracture (LIENERT *et al.*, 2011).

Non-destructive inspection (NDI) techniques evaluate the quality of the welded joint without damaging or altering it. NDI methods are more common than destructive testing because they are less expensive, less time-consuming, and more convenient. As observed by Thompson and Chimenti (2012), the most common NDI methods are visual testing, radiography, ultrasonic testing, magnetic particle testing, and liquid penetrant testing. Visual testing by a human specialist is the simplest and most cost-effective NDI method, as it requires a trained inspector to visually examine the weld. Radiography uses X-rays or gamma rays to create an image of the welded joint, which can reveal internal defects. Ultrasonic testing employs high-frequency sound waves to detect flaws or discontinuities within the joint. Magnetic particle testing detects surface and near-surface defects by applying a magnetic field and observing the behavior of magnetic particles. Liquid penetrant testing identifies surface defects by applying a penetrating fluid that seeps into cracks or pores, which is then removed before the part is examined under ultraviolet (UV) light.

Proposed methods closely related to the inspection of weld defects range from those depending on multiple sensors (audio and electric current) (HONG *et al.*, 2023b; ZHAO *et al.*, 2022; MIAO *et al.*, 2019a); to more specific sensors, such as laser (MIAO *et al.*, 2022), ultrasound (POLIKAR *et al.*, 1998) and spectrum components (ZHANG *et al.*, 2019). An early method was even proposed to predict failures by learning welding process parameters (ZENG *et al.*, 1993). In particular, we focus on methods that rely on image processing and machine learning, which are mostly NDI techniques. We categorize the methods according to the main techniques used – handcrafted features or deep learning features – and according to the processed data – mostly X-ray and visible spectrum images. As shown in Figure 6, handcrafted methods were dominant before 2020. Since then, deep learning techniques prevail, with NDI methods using CNNs (Convolutional Neural Networks) and derived architectures, such as GANs (Generative Adversarial Networks) and Transformers. Table 1 provides a detailed overview of the most recent

Figure 6 – Timeline of NDI methods for weld defects detection and classification. Source: Author.



and relevant works that employ images as input data. To clearly present and substantiate our contributions, some of those works are detailed as follows.

We initially consider works based on X-ray data to analyze Table 1 more objectively. A wide range of datasets and classified faults can be observed, from significantly reduced (e.g., (YANG *et al.*, 2021)) to more extensive datasets (e.g., (GUO *et al.*, 2021)). The studies by (KUMARESAN *et al.*, 2021a) and (YANG *et al.*, 2022) stand out, as they classify up to 14 types of faults; and in (YANG *et al.*, 2022), the defect regions are segmented as well. Defects such as cracks, lack of fusion, and lack of penetration are the most evaluated. This may be related to these defects being more evident in industrial radiographs (NACEREDDINE *et al.*, 2019).

Table 1 – Overview on welding defects classification using images. *mAP for low and high-resolution images considering a YOLOv7-tiny architecture for detection and classification of the four defect types. Source: Author.

Paper	Year	Image	Dataset size	Defect classes	Number of defects	Architecture	Public?	Results
(YANG *et al.*, 2021)	2021	Radiography	20	defective/non-defective	20	Unet and GAN	Yes	88.4% (Acc)
(GUO *et al.*, 2021)	2021	Radiography	20,360	crack, lack of fusion, lack of penetration, slag inclusion, porosity, normal	4,640	YOLO and GAN	No	90.9% (F1)
(CHANG; WANG, 2021)	2021	Radiography	NA	background, slag inclusion, porosity, crack	NA	Custom SegNet	Yes	98.6% (Acc)
(PATIL; REDDY, 2023)	2021	Radiography	80	crack, undercut, gas pores, porosity, slag, warm holes, lack of penetration, non-defective	NA	handcraft and ANN	No	98.75% (Acc)
(KUMARESAN *et al.*, 2021a)	2021	Radiography	940	14 different types	940	CNN and transfer learning	Yes	98% (Acc)
(LIU *et al.*, 2022b)	2022	Radiography	477	burn through crack, lack of penetration, lack of fusion, bar, concave, round defect	357	Sup-Con	No	91.98% (Acc)
(YANG *et al.*, 2022)	2022	Radiography	940	14 different types	940	Custom Unet	Yes	85.4% (dice)
(LIU *et al.*, 2022a)	2022	Radiography	3,000	pores, slags, lack of fusion, lack of penetration, crackles, undercuts	2,714	Resnet and attention	No	85.4% (mAP)
(XU *et al.*, 2022)	2022	Radiography	5,852	crack, porosity, slag inclusion, lack of penetration, lack of fusion	5,852	FPN-ResNet-34	Partially	73% (mIoU)
(YANG *et al.*, 2023)	2023	Radiography	88	cracks	NA	CNN custom Unet	Yes	96.2% (Acc)
(TRIPICCHIO *et al.*, 2020)	2020	Visual	7,600	blow Hole, lack/excess material, misalignment, thin/large joint	340	DNN Custom DenseNet	No	96.3% (Acc)
(SKILTON; GAO, 2020)	2020	Visual	969	defective/non-defective	NA	YOLO and GAN	No	84% (AUC)
(GAO *et al.*, 2020)	2021	Visual	1,800	crazing, inclusion, patches, pitted, rolled-in, scratches	1,800	CNN based on Resnet50	Yes	41.8% (mAP)
(DENG *et al.*, 2021)	2021	Visual	5,200	gas pores, cracks, lack of fusion, lack of penetration	3,200	CNN VGG16	No	98.75% (Acc)
(WANG *et al.*, 2022)	2022	Visual	42,229	incomplete fusion, partial, full, excessive penetration	42,229	Transformer	No	98.11% (Acc)
(NOWROTH *et al.*, 2022)	2022	Visual	282	pores, cracks, shape defect	NA	Custom Xception	No	95% (Acc), 76.88% (mIoU)
(HONG *et al.*, 2023a)	2023	Visual	2,219	undercut, snake-like, sound	2,219	ROI and handcraft	No	96.72% (Acc)
(HONG *et al.*, 2023b)	2023	Visual	1,400	lack of fusion, humping, sound	1,400	ROI and handcraft	No	94.98% (Acc)
LoHi-WELD (**Proposed**)	2024	Visual	3,022	pore, deposit, discontinuity, and stain	22,412	YOLOv7	Yes	64% (mAP)* 69% (mAP)*

A notable aspect of these works is the existence of a public dataset for benchmarking results: GDXRay (MERY *et al.*, 2015). This dataset covers a wide range of defects. However, it suffers from a significant class imbalance and contains relatively few images (fewer than 200 per class), which poses challenges for deep learning models.

Despite relevant results on the fronts of detection, classification, and segmentation, methods based on X-Ray images have some limitations, such as (i) the need for specialized equipment to acquire radiographic images on the production line; (ii) a primary focus on mitigating the effects of dataset imbalance, which is prevalent in most studies; (iii) although these methods often provide detailed comparisons with the literature, they are largely restricted to the public GDXRay dataset; and (iv) there is a lack of initiatives to build publicly available, more comprehensive, and less unbalanced datasets.

In visual methods using grayscale or RGB images, both handcrafted and deep learning-based approaches have achieved state-of-the-art results. While the number of defects evaluated in these methods is generally lower than in X-ray-based approaches, the ease of data acquisition often results in a higher average number of images. Similar to methods based on radiography, there are few publicly available datasets. Notable datasets such as NEU-DET, DeepPCB, KolektorS, DAGM2007, and Real-world Glass Bottle, referenced in (WANG *et al.*, 2023), are not specifically designed for weld defects. Others, as seen in (GAO *et al.*, 2020), are tailored to specific industrial applications, such as printed circuit boards.

For handcrafted methods, one can emphasize the approach presented in (HONG *et al.*, 2023a) and (HONG *et al.*, 2023b). Different image processing stages are applied to the image before extracting the features for the classifiers. The main limitation of these methods is the specificity of the solution, which is typically well-adjusted for a particular image acquisition scenario, thus limiting the generalization of the model for noisier scenarios or for different conditions of data collection. In addition, preprocessing considers specific aspects of the collected images and cannot always be used in other contexts.

On the other hand, CNN-based methods can improve generalization and present more generic solutions for different lines and products. In (TRIPICCHIO *et al.*, 2020), the authors presented methods based on DenseNet to detect and classify welding defects, known as blow holes, lack/excess of material, misalignment, and thin/large joints. The methods were expected to be adaptable to changes in the production line. They reported achieving an accuracy of 96.30% in classifying defects, even though the authors needed to add transfer learning and handcrafted techniques, such as image filtering, to help overcome retraining samples' limitations due to scenario changes. Similarly, in (NOWROTH *et al.*, 2022), the researchers proposed a semantic segmentation approach for weld contours. The research includes detecting weld metal, background, and defects (cracks and pores) using a pre-trained neural network. The study produced a high-definition dataset containing 282 images for semantic segmentation models.

Deep learning methods have recently focused on GANs and Visual Transformers (ViTs). Skilton and Gao (2020) proposed to use a GAN to detect anomalies — patterns in data that do not conform to a well-defined notion of normal behavior, as defined in (CHANDOLA *et al.*, 2009) — in a nuclear-fusion experiment known as JET (Joint European Torus), which may suffer from weld cracks, melting, and debris, where the goal was to model the normal behavior of samples using an adversarial training and detecting the anomalies by using an anomaly pixel-wise score. A similar approach was recently employed in (WANG *et al.*, 2023). The ViT model was used by (WANG *et al.*, 2022) for the task of automatic welding penetration recognition in a robotic system. The authors trained ViT models from scratch with different architectures and showed the influence of model parameters on the recognition performance.

They also used transfer learning from ViT architectures trained on ImageNet to address the issue of modeling complexity and lack of training data. The ViT model outperformed other CNN architectures in classifying four penetration states: incomplete fusion, partial penetration, full penetration, and excessive penetration. The authors tested their ViT model on 42,229 grayscale weld pool images. Hybrid methods that combine wavelet features with CNNs are also described (MIAO *et al.*, 2019b).

Although recent approaches point to visual images and deep learning methods, currently available public datasets still lack real and comprehensive cases to validate weld defects detection and classification properly. Furthermore, none of them have a specific annotation for detecting defects in images. Finally, resolution issues for identifying defects still need to be addressed and detailed in recent work on this subject. In this sense, this work's main contribution is to provide a public dataset that serves as a reference for detecting and classifying weld defects, proposing two sets of data collected with different resolutions. Additionally, we present a baseline with state-of-the-art object detection methods based on deep learning, pointing out possible paths and areas for improvement for a practical and robust application in this field.

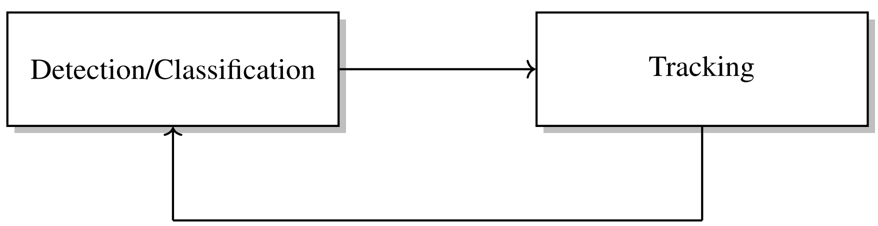
3 METHODOLOGY FOR IMPRINT DEFECTS

In this chapter, we discuss the proposed methodology used for the detection and classification of imprint stamping defects. As explained in Section 1.2, these defects manifest as peak-like formations and are categorized into mild imprints — defects that can be repaired through subsequent polishing and painting — and severe imprints — defects that are irreparable and necessitate the disposal of the parts. These defects are analyzed through a sequence of images captured as the stamped parts move through the inspection chamber.

3.1 FRAMEWORK OVERVIEW

The main problem during the recognition of imprint defects is that the movement of the produced part in the inspection chamber, as well as the camera and light arrangement setup, highly influence the image acquisition. Thus, it may be easier or harder to find an imprint defect depending on the location of the defect in the manufactured part, and the location of the part during the treadmill slide. To cope with these problems, we use a sequence of images, instead of a single image, to detect and classify the imprint defects. The proposed framework for this problem is designed to combine detection, classification, and tracking in a synergistic way, as shown in Figure 7.

Figure 7 – Proposed framework overview: the recognition of imprint defects relies on the detection, classification and tracking of defects along a sequence of frames. Source: Author



By matching detected and tracked regions over a sequence of image frames, it is possible to consider a voting analysis to produce a more accurate response than that obtained using independent detections on every single frame. Algorithm 8 outlines the proposed surface inspection pseudo-code.

The algorithm receives as input a sequence of n consecutive images (frames) $\mathbb{F}^{(1)}$, $\mathbb{F}^{(2)}$, ..., $\mathbb{F}^{(n)}$. Step 4 crops an image frame into m overlapping sub-images of equal size. As explained

Figure 8 – Algorithm of Surface-Inspection ($\mathbb{F}, n$). Source: Author.

input : A sequence of n image frames $\mathbb{F} = \{\mathbb{F}^{(1)}, \ldots, \mathbb{F}^{(n)}\}$
output: A list L of trajectories for potential surface defects.

1 $L \leftarrow \{\}$; $\triangleright$ List of regions trajectories
2 $T \leftarrow \{\}$; $\triangleright$ List of regions being tracked
3 **for** $i \in \{1, 2, \ldots, n\}$ **do**
4     $\{\mathbb{R}^{(1)}, \ldots, \mathbb{R}^{(m)}\} \leftarrow$ Image-Cropping ($\mathbb{F}^{(i)}$) ;
5     $\{\bar{D}^{(1)}, \ldots, \bar{D}^{(m)}\} \leftarrow$ Detect-and-Classify ($\mathbb{R}^{(1)}, \ldots, \mathbb{R}^{(m)}$);
6     $\bar{D} \leftarrow$ Region-Mapping ($\bar{D}^{(1)}, \ldots, \bar{D}^{(m)}$) ;
7     $D \leftarrow$ Non-Maxima-Supression ($\bar{D}$) ;
8   **for** $d \in D$ **do**
9         $t \leftarrow$ Intersection (d,T) ;
10   **if** object t is empty **then**
11             $T \leftarrow T \cup d$; $\triangleright$ Add a new detection for tracking
12   **else**
13             $t[\text{ndetections}] \leftarrow t[\text{ndetections}] + 1$;
14             $t[d[\text{class}]] \leftarrow t[d[\text{class}]] + 1$;
15   **end**
16   **end**
17     $\{T, \hat{T}\} \leftarrow$ Track-Regions (T);
18     $L \leftarrow L \cup$ Analyze ($\hat{T}$);
19 **end**
20 **return** L;

in Section 3.2, this avoids image resizing and consequently preserves high-frequency content necessary to detect imprint defects. Then, a convolutional neural network is fed with each of the m sub-images and produces as output a list of candidate detection regions along with the associated classifications (step 5). The candidate regions are mapped to the original uncropped image, and duplicated occurrences are eliminated using non-maxima suppression (steps 6 and 7).

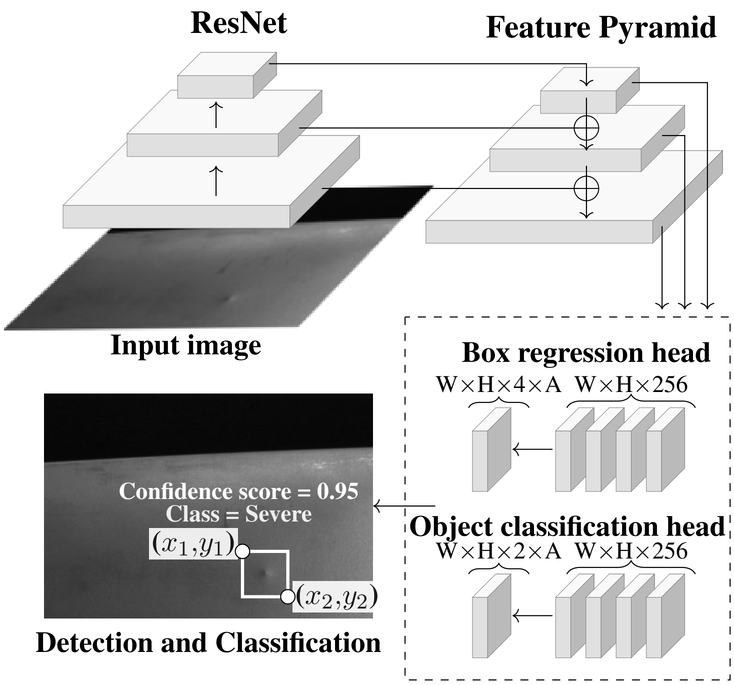
The loop in step 8 matches detected regions with tracked regions (step 9). Whenever the algorithm fails to find a match (step 10), a new tracker is created with a unique identifier and is added to the tracking list T (step 11). For those regions already being tracked, we accumulate in tracker t the number of times the region was re-identified (step 13), and the number of occurrences for each class, namely, mild or severe defect (step 14). These information are later used to check the consistency of the detection and classification.

In step 17, the regions in T are tracked in the next frame. This procedure generates two sets, one composed by those regions in T successfully tracked, and other $\hat{T}$ with the trajectories of objects interrupted either by occlusion or by any tracking failure. At this point, we can use the knowledge obtained from detection/classification and tracking (step 18) to assign a most likely category for the region. It is also possible to determine whether a detected region is a false alarm or a true defect by analyzing the length of the tracked sequence. In the following, we provide further details on the key steps of the algorithm.

3.2 SURFACE DEFECT DETECTION AND CLASSIFICATION

The convolutional neural network used for detection and classification is known as RetinaNet (LIN *et al.*, 2017), a state-of-the-art architecture composed of a Feature Pyramid Network on top of a Residual Network (ResNet) (HE *et al.*, 2016), see Figure 9.

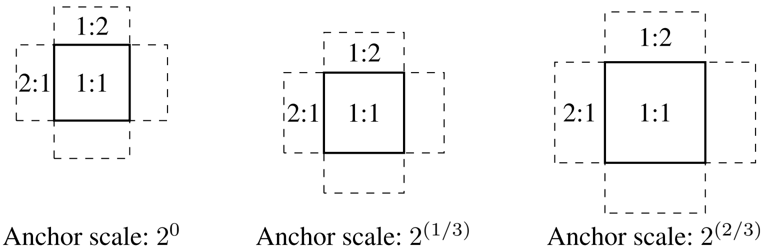
Figure 9 – Scheme of a Retina Network, as proposed by Lin *et al.* (LIN *et al.*, 2017), exemplified with three scale levels for the feature pyramids. The input is an image of height H and width W. The box regression head predicts the pixel coordinates $\{x1,y1,x2,y2\}$ for A anchor boxes (region proposals). The object classification head predicts the class and a probability score for each object region. We show only 3 pyramid levels. Source: Author.



The ResNet pyramid generates deep features at different scales to compose a feature pyramid. The initial feature map comes from the top of the ResNet. The next level is computed by element-wise addition of an upsampled version of this initial map and the feature map from the same scale in the ResNet pyramid. Each subsequent level is computed following this top-down approach. RetinaNet defines five levels which are named from $P3$ to $P7$.

The feature maps are used by two sub-networks. The first extracts axis-aligned rectangular bounding boxes $r = (x_1,y_1,x_2,y_2)$ — (x_1,y_1) is the upper left corner and (x_2,y_2) is the bottom right corner — corresponding to candidate imprint defects. The boxes are based on anchors with areas from 32^2 to 512^2 on pyramid levels from $P3$ to $P7$. Each level samples nine anchors with three aspect ratios (1:2, 1:1, 2:1) and three scaling factors (2^0, $2^{(1/3)}$, $2^{(2/3)}$), see Figure 10. The second sub-network classifies candidate regions (each of the nine anchors) into mild or severe defects, providing a confidence score for the classification expressed as a real number within the interval $[0,1]$.

Figure 10 – Anchor boxes used by RetinaNet to detect objects. Source: Author.



A deep neural network traditionally rescales the input image to a standard dimension for training and inference. However, scaling a high-resolution image into blocks of 600×600 or 800×800 pixels, for example, may remove fine-details, such as the texture that characterizes a mild imprint, and compromise the recognition. Therefore, instead of scaling, we crop the input image into sub-images with the same dimensions expected by the convolutional neural network. Furthermore, the cropping operation is performed with 200 pixels overlap to avoid missing imprint defects located at the borders (Figure 11).

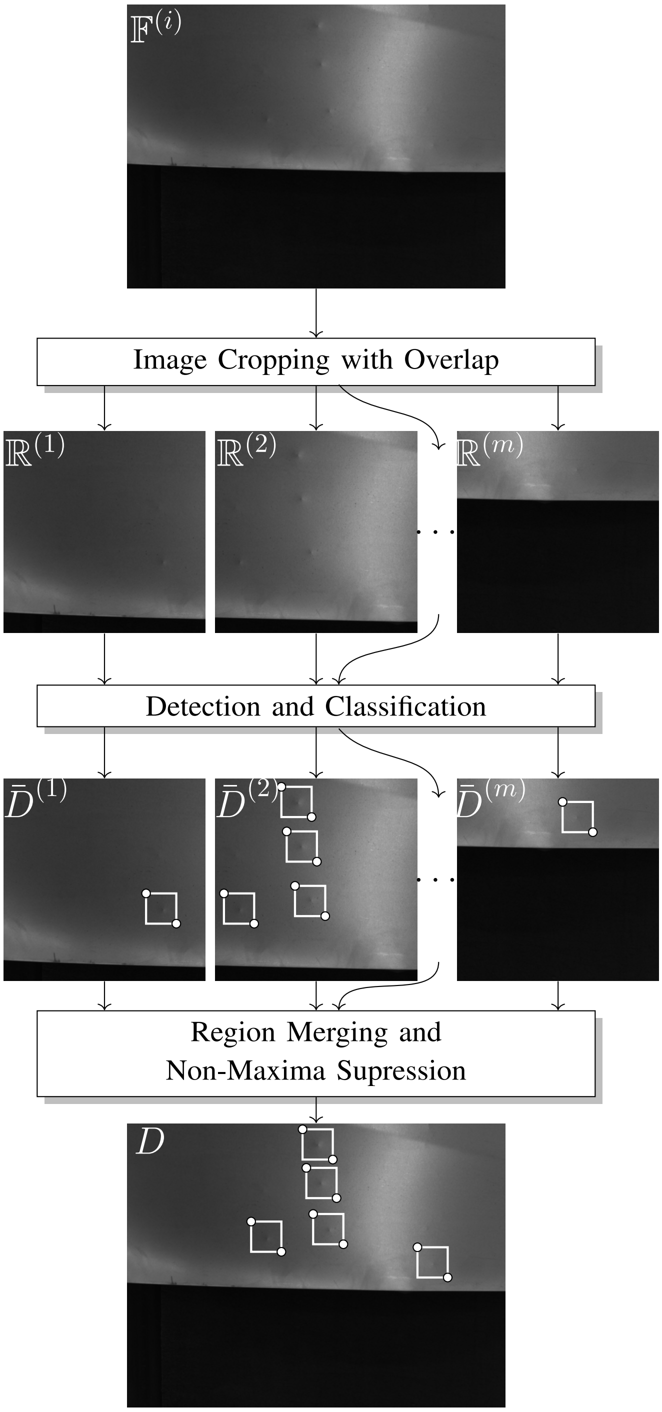
For training, we feed the RetinaNet with cropped images, of 800×800 pixels, as well as a list of imprint defects, which are defined by a bounding box with the coordinates of a surface defect and the corresponding classification — reported by a specialist.

The detections found for each sub-image are remapped to the corresponding patches in the original image. Since the same defect may be detected multiple times due to region overlapping, we apply non-maxima suppression to preserve just those regions with the highest scores.

3.3 TRACKING

For surface inspection of imprint defects it is not sufficient to apply a detector independently to each image frame. This simplistic approach ignores the temporal coherence between adjacent frames and produces uncorrelated detections for the same defect, as illustrated in Fig-

Figure 11 – Detection pipeline flowchart: the input is a high-resolution image and the output is a set of detected regions with associated classification. Source: Author.



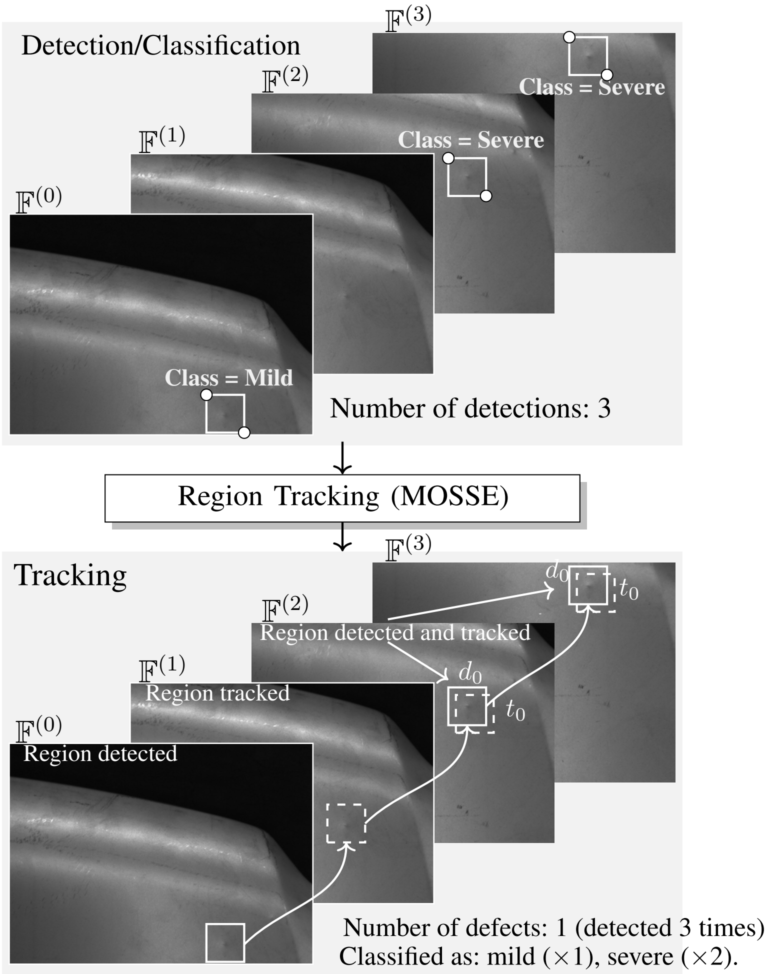
ure 12 (top). Furthermore, light reflection differences over time may change the classification label of a region or cause detection failures. In this context, the tracking step allows:

- exploration of the temporal redundancy during the inspection process to find missing detections which supposedly correspond to the same defect;

- assessing the detection consistency by computing the number of occurrences of a given imprint defect in the tracked path;
- assessing the classification consistency by observing the classifications assigned to an imprint defect in the tracked path.

Figure 12 (bottom) illustrates how tracking is used to improve the detection accuracy.

Figure 12 – Extraction of tracking sequences to asses the coherence of detection and classification. Source: Author.



For a sequence of n images $\mathbb{F}^{(0)}$, $\mathbb{F}^{(1)}$, …, $\mathbb{F}^{(n-1)}$, let $D \in \mathbb{F}^{(i)}$ be a list of detected regions and $T \in \mathbb{F}^{(i)}$ a list of tracked regions. We compute matchings between D and T by using the *intersection over union* (IoU)

$$\text{IoU}(d_i, t_j) = \frac{\text{area } (d_i \cap \ t_j)}{\text{area } (d_i \cup \ t_j)}, \tag{1}$$

where $d_i \in D$ and $t_j \in T$. If the IoU value is at least σ (set to 0.5 in our experiments), the regions are considered a match. We create a new tracker for each detected region in D that does not match any tracked region in T. We also keep tracking in the subsequent frames regions that do not match any region of D.

When a match is found for d_i and t_j, the region being tracked is replaced by d_i, i.e., the matching strategy implicitly assumes the predicted regions are less accurate or less reliable than those detected by the network. The predicted regions are used only to identify regions belonging to the same detection. We add to t_j a counter to compute how many times the region is detected — the higher the indicator, the more likely is the region to contain an imprinting defect. We also collect the classes assigned to each region throughout a tracking sequence to perform a majority voting and decide the most likely classification. That is important since the assignments can alternate between mild and severe defects, mainly due to differences of light reflection on the metal surface.

We used in our framework the Minimum Output Sum of Squared Error (MOSSE) tracker, developed by Bolme *et al.* (2010). MOSSE is a filter-based tracker that receives as input the targets found by the detector and follows them in the subsequent frames by correlating a filter over a search window. For reasons of efficiency, the tracking occurs in the frequency domain by using the Fast Fourier Transform. A good tracking response for MOSSE produces a peak in the frequency domain corresponding to the target center. To measure the peak strength and to determine a tracking failure the authors used the Peak to Sidelobe Ratio (PSR). We observed in our problem that a PSR of 5.0 or lower is an indicator of tracking failure.

4 EXPERIMENTAL RESULTS FOR IMPRINT DEFECTS

In this section, we present the image acquisition machinery, the dataset produced, and the experimental evaluation of the proposed method for imprint defects.

4.1 IMPRINT DEFECT DATASET

Figure 13 shows the image inspection system, made available by an automotive industry, to obtain the image dataset for our experiments. The system contains a set of cameras, a hybrid system of high-frequency fluorescent lamps combined with bright white LED strips, two laser sensors, an encoder, an electric panel with a programmable logic controller (PLC), and two dedicated computers for image processing. Cameras attached to the system's chamber, that can capture the entire surface of the inspected parts, were used to acquire the image sequences for our dataset, see Figure 14.

Figure 13 – Visual inspection system used by an international automotive industry, our partner in this project, for surface defect detection. Source: (BLOCK, 2018).

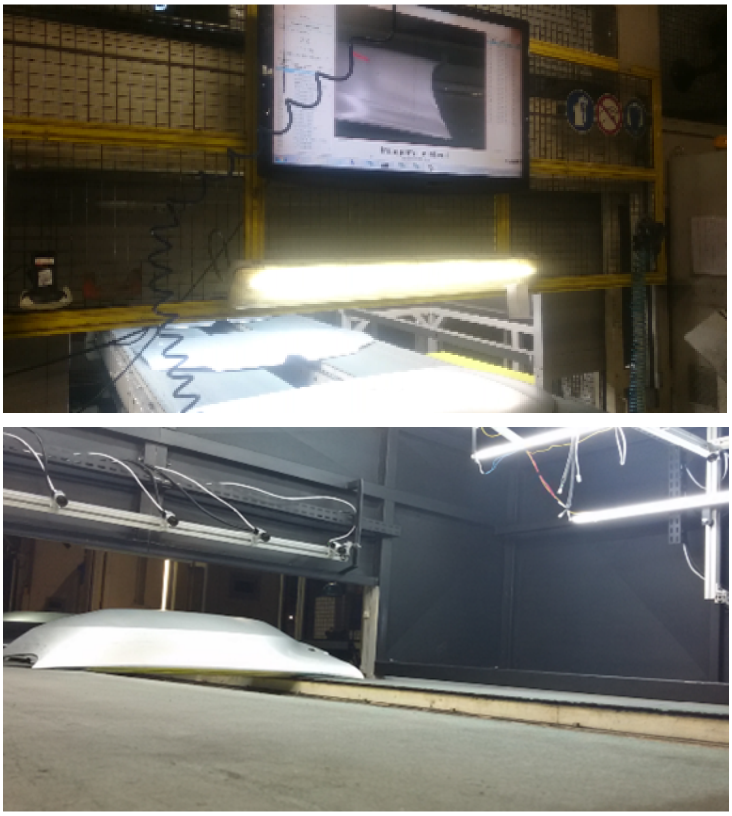
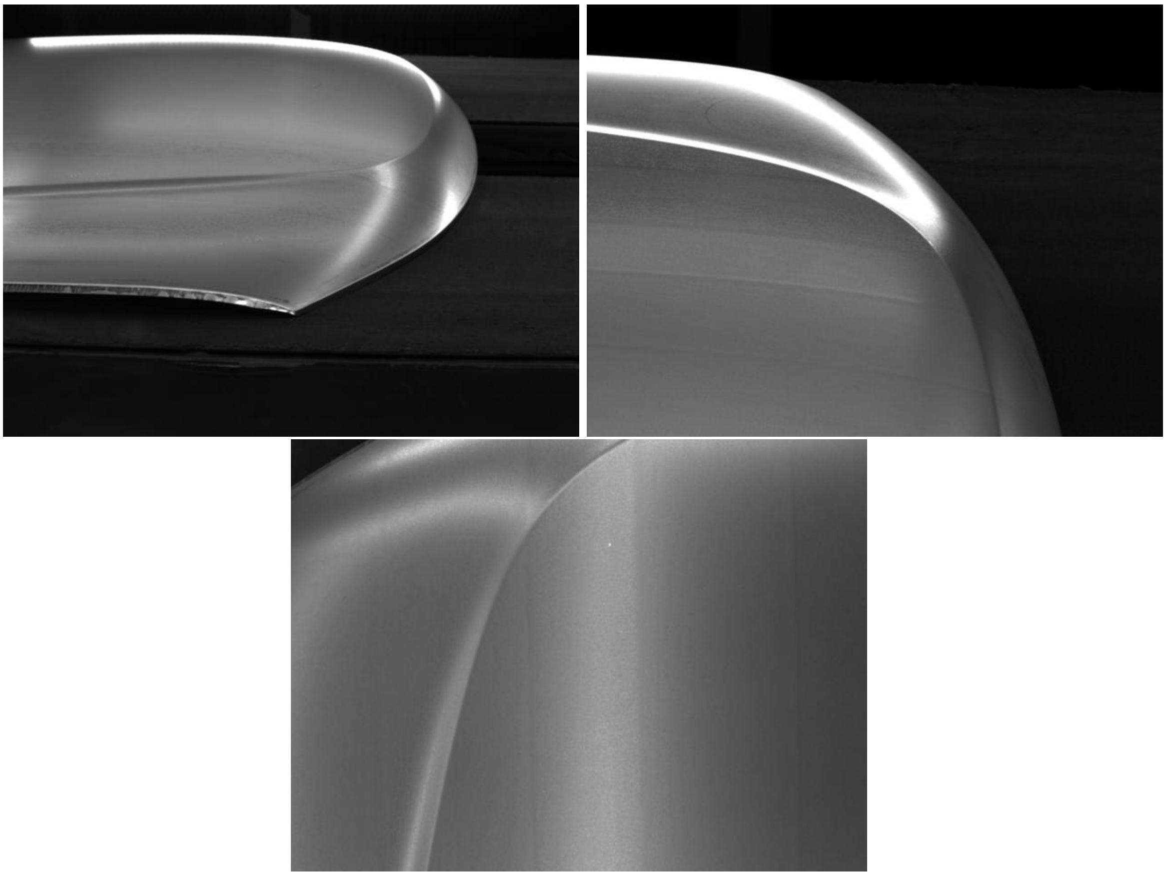


Figure 14 – Image samples collected from the visual inspection system. Source: Author.



Since the source of imprint defects must be removed as soon as possible, it is difficult to obtain, from the daily production line, the required amount of images for training. Therefore, with the help of an expert, a punching tool was used to create defect samples equivalent to those coming from a press machine.

We manually annotated the image sequences to produce the ground truth with the help of a quality control expert who physically inspected each existing imprint defect. Each annotated defect is represented by the pixel coordinate indicating the peak center, the class (mild or severe), and a unique identifier, used throughout the whole video for a same physical defect. Table 2 summarizes the dataset (number of videos, number of images, and number of marks in the ground truth for severe and mild defects).

Table 2 – Dataset and the partition size of the training and testing subsets. Source: Author.

	# Videos	**# Images**	**# Mild**	**# Severe**
Training	8	5,594	6,532	2,313
Testing	23	8,481	6,220	2,364
Total	31	14,075	12,752	4,677

4.2 DETECTION AND CLASSIFICATION RESULTS

In this section, we initially present a comparison of RetinaNet against a detector known as *You Look Only Once* (YOLO) (REDMON *et al.*, 2016) — a real-time object detector, also based on convolutional neural networks, that achieved outstanding results in the Pascal Visual Object Classes (VOC) (Everingham *et al.*, 2010) — and then we evaluate the detection and classification results of RetinaNet.

To measure the detection robustness, we process all image frames $\mathbb{F}^{(i)}$ for $i = \{0, 1, 2, \ldots, n-1\}$, and compare the set $D^{(i)}$ of detected regions with the ground truth $G^{(i)}$. A detection is considered valid if the classification confidence is at least 0.5. A detected region that has an overlap (based on the IoU value) — see Equation (1) — of at least τ with a ground truth region is considered a true positive (tp), otherwise, it is considered a false positive (fp). A region is also considered a false positive when the class assigned to it differs from the ground truth. Ground truth regions which were not detected are considered false negatives (fn). We set τ as 0.5 in our experiments.

In our experiments we used the precision and recall metrics. The precision (P) is given by

$$\mathrm{P} = \frac{|\mathrm{tp}|}{|\mathrm{tp}| + |\mathrm{fp}|} \tag{2}$$

such that $|\mathrm{tp}|$ is the number true positives and $|\mathrm{fp}|$ is the number of false positives. Similarly, the recall (R) is given by

$$\mathrm{R} = \frac{|\mathrm{tp}|}{|\mathrm{tp}| + |\mathrm{fn}|} \tag{3}$$

such that $|\mathrm{fn}|$ is the number of undetected defects (false negatives).

We also used the Mean Average Precision (mAP) metric, as defined in the PASCAL Visual Objects Classes (VOC) challenge (Everingham *et al.*, 2010). The Average Precision (AP) for a class is based on the area under the curve of the precision/recall curve. Initially, we order the detections by their confidence and rank the results to evaluate the trade-off between false positives and false negatives. Recall and precision are, respectively, the proportion of all positive examples above a given rank and the proportion of all examples above that rank from the positive class. The interpolated average precision is computed by numerical integration of the approximate curve with monotonically decreasing precision (the precision for a given recall r is set to the maximum precision for any recall greater than r). Therefore, the mAP metric is the mean of the average precisions for all classes.

Table 3 shows the results obtained for YOLO, RetinaNet, an ensemble of YOLO plus RetinaNet, and for the proposed framework with RetinaNet and MOSSE. The table reports the mAP and the APs (for mild and severe defects). We trained the networks YOLO (version 2) (REDMON; FARHADI, 2017) and YOLO (version 3) (REDMON; FARHADI, 2018). The former yielded much better results. The YOLO + RetinaNet ensemble results come from the combination, using non-maxima suppression, of the YOLO and RetinaNet detections.

Table 3 – Performances of individual state-of-the-art detectors (YOLO and RetinaNet) and their ensemble (YOLO + RetinaNet) against the proposed framework by using the Average Precision (AP) and mean Average Precision (mAP) metrics, with a minimum region overlap of 50%. Source: Author.

Algorithm	Severe (AP)	Mild (AP)	mAP
YOLO (version 2)	81.53%	39.46%	60.50%
RetinaNet	83.78%	47.46%	65.62%
YOLO + RetinaNet (ensemble)	85.43%	49.82%	67.62%
Proposed framework	**88.63%**	**63.80%**	**76.21%**

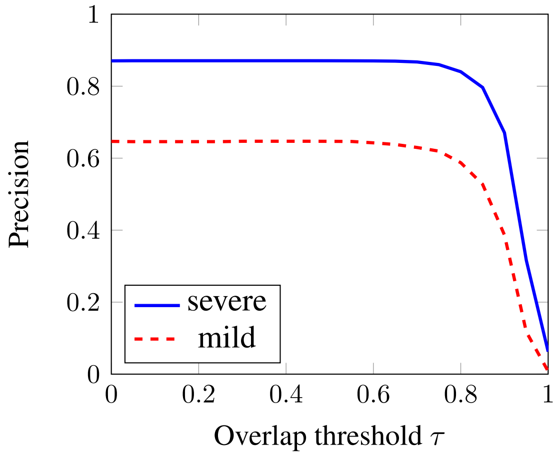
RetinaNet outperformed YOLO, mainly for mild defect classification. The ensemble YOLO + RetinaNet did not obtain a significant improvement when compared to RetinaNet itself. The ensemble was designed to evaluate whether the combination of the networks would produce complementary detections. However, that does not seem to be the case, and there was only a small gain at the expense of a much higher computational cost. Therefore, we decided to explore RetinaNet to report the results obtained with the addition of the tracking step in our framework.

The results of the proposed framework in Table 3 also reveal the effectiveness of exploring the coherence among tracked regions to improve detection. Since a defect can be detected in many frames, possibly with different class labels, we used the tracking sequences to correlate the many detected instances and give a final class label according to the most common label throughout the sequence (set using majority voting). We also use the tracking sequence to weight the confidence of the RetinaNet detections.

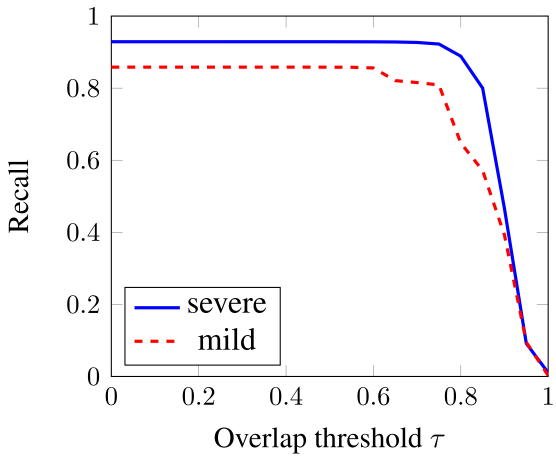
We now explore the tracking influence on the classification for the combination of detection/classification + tracking. Figures 15(a) and 15(b) show the precision and recall curves for overlap thresholds τ in the range $[0,1]$. Observe that until an overlap of 0.6, the method achieves closely the same precision and recall values. The detection of severe defects yielded a precision of 0.87 and recall of 0.93, while mild defects presented a precision of 0.65 and recall of 0.86.

The tracked regions can be used to improve precision. The amount of false positives

Figure 15 – Precision and recall for the detection step with varying overlap thresholds. Source: Author.



(a)



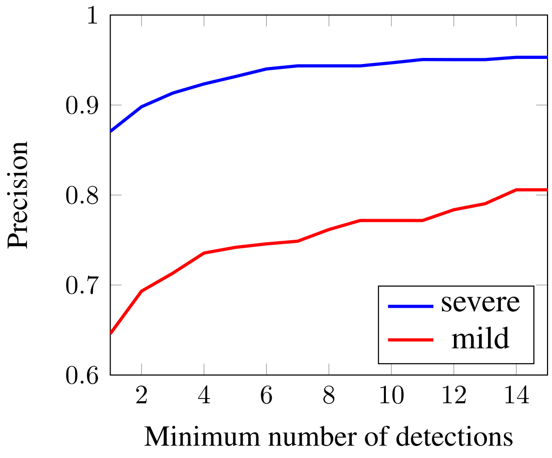
(b)

for mild defects can be high due to noisy false detections resulting from the inherent difficulty of this type of defect. We noticed the number of detections in the tracked sequences is low in such cases and, as a consequence, we are able to impose a restriction on the minimal number of detections to consider a true detection. Figures 16(a) and 16(b) show the results obtained when forcing a minimal number of detections for each tracked region.

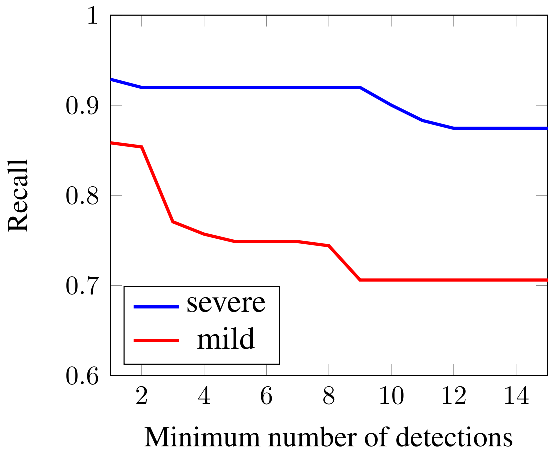
By simply removing sequences with only one detection, the precision and recall values for severe defects are 0.90 and 0.92, respectively. For mild defects, the values are 0.69 and 0.85. Since mild defects are not critical and can be fixed along the production line, it is possible to use independent machinery for mild defects and avoid false alarms (at the expense of a lower recall).

Figure 17 depicts how the combination detection/classification + tracking helps to detect true defects. The first row shows the detections obtained by RetinaNet. Most regions are unstable, appearing and disappearing in adjacent frames, except for that at the bottom right, detected in the whole sequence. That is the type of behavior explored by the combination of detection and

Figure 16 – Precision and recall given a required number of detections. Source: Author.



(a)



(b)

tracking, where the temporal coherence among frames is explored to clean unstable regions and keep the true imprint defect.

In a final experiment we performed an analysis of RetinaNet's parameters to evaluate the trade-off between detection/classification performance versus computational time, see Table 4. The detection/classification step is the most time consuming part of our framework. The following four changes in RetinaNet were used in the experiments: (1) use of only the 1:1 aspect ratio; (2) use of only the 2^0 scale; (3) use of two pyramid levels, P3 and P4; and (4) use of three pyramid levels, P3, P4 and P5. As can be seen, there are no significant gains in terms of both time and performance. The experiments were carried out on an Intel Core i7, 64 GB of RAM, running Linux and Python, with a GPU Titan Xp (12 GB). We trained the architectures with 10 epochs using Adam optimization (initial learning rate of 1e-5).

Figure 17 – The recognition of a true severe defect (highlighted on the right) during the inspection: (top) from a large number of false detections only one refer to a true defect; (bottom): we filtered out the trajectories of those regions that were not constantly re-detected and regularly assigned to the same class. Source: Author.

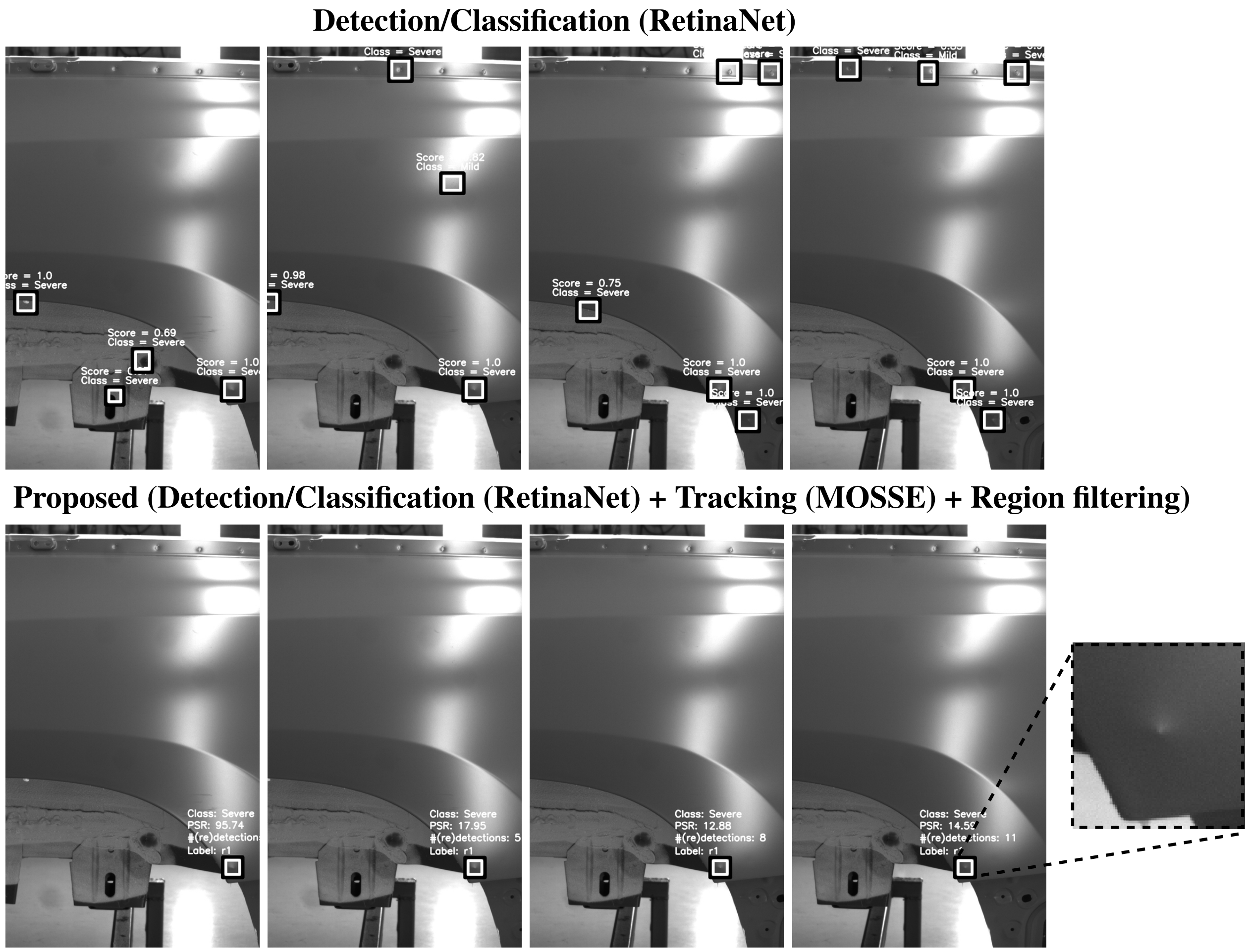


Table 4 – The performance of the proposed framework by using four versions of the original RetinaNet network as an attempt to reduce the computational time. Average execution time per image for the whole framework. Source: Author.

Proposed framework	Severe (AP)	Mild (AP)	mAP	Time (s)
+ RetinaNet (ratio 1:1)	88.19%	60.32%	74.25%	0.190
+ RetinaNet (scale 2^0)	86.24%	52.95%	69.59%	0.191
+ RetinaNet (pyramid P3-P4)	87.48%	59.48%	73.48%	**0.173**
+ RetinaNet (pyramid P3-P5)	**90.55**%	62.18%	**76.37**%	0.179
+ RetinaNet (original)	88.63%	**63.80**%	76.21%	0.193

4.3 DISCUSSION

Imprint defects on stamped metal surfaces are challenging to detect, and techniques specifically designed for their detection and classification are scarce in the literature. We believe that this field presents several open challenges that we discuss in the following.

- Enhancing mild defect detection: Achieving higher precision in the classification of mild defects remains a significant challenge, as these are often harder to detect and prone to generating false positives.

- Real-time tracking in complex environments: Leveraging SORT (Simple Online and Real-time Tracking) algorithms, which utilize features extracted directly from convolutional neural networks, can accelerate the tracking process. Techniques such as those proposed by (WOJKE *et al.*, 2017; DU *et al.*, 2023) may show promise in this area.

- Addressing lighting variability: As highlighted by (REN *et al.*, 2022), many computer vision systems employ structured light architectures to emphasize key surface characteristics during inspection. Applying such techniques could enhance the visibility of imprint defects on metal surfaces, improving detection and classification accuracy.

- Integration with industrial systems: Integrating the proposed framework into existing industrial quality control systems, along with automated feedback mechanisms, could enhance real-time decision-making processes and overall production efficiency.

5 METHODOLOGY FOR WELDING DEFECTS

In this chapter, we present a framework designed to detect and classify welding defects in digital images captured during the automotive assembly line, aiding in non-destructive visual inspection. The proposed dataset, detailed in Section 5.1, includes images of four types of defects — porous, deposits, discontinuities, and stains — and is crucial to the framework's development. Decisions discussed in Section 5.2, such as the architecture for defect detection and classification and the strategies for image data augmentation, are significantly influenced by the image acquisition process, including factors like camera types and lighting conditions.

5.1 WELD DATASET

The weld bead images are obtained from a structural base of the vehicle known as "monocoque", as shown in Figure 18. We compiled two datasets acquired under distinct light exposure (there is not a closed chamber in the robots area and the captured images are subject to variations in lighting), image resolution, and camera configurations for this project. These datasets, named here as `weld-low` and `weld-high` , were automatically recorded by two distinct cameras. The `weld-low` dataset consists of $2{,}000$ weld bead images collected with a low-resolution image sensor (640×480 pixels). In contrast, the `weld-high` dataset consists of $1{,}022$ weld bead images collected with a high-resolution image sensor (2048×1080 pixels). For both datasets, the image acquisition was performed by using a camera with a CMOS sensor and a global shutter, in gray-scale, encoded in JPEG image format. A LED panel with white light as illumination was used to ensure adequate sharpness for the inspection of the weld quality. As the goal is defect detection and classification, we properly oriented and cropped the weld beads from the original image frames.

As shown in Figure 19, the `weld-low` dataset was obtained focusing on two distinct welding regions, one with a width dimension of $\approx$ 40 mm, and another one larger, with $\approx$ 60 mm, as detailed in Figure 21a. We collected 1,000 distinct occurrences of each one in a series of experiments. For our experiments, both weld beads were kept horizontally aligned, with rotation applied if necessary to achieve this alignment.

The `weld-high` dataset is composed only of the larger weld bead, see Figure 20,

Figure 18 – The weld beads used in this research are located in the vehicle monocoque region highlighted in red. Source: the monocoque image is available in Wikimedia commons.

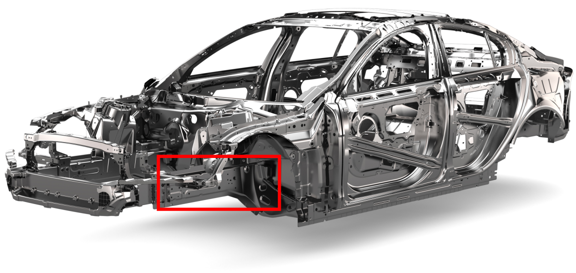
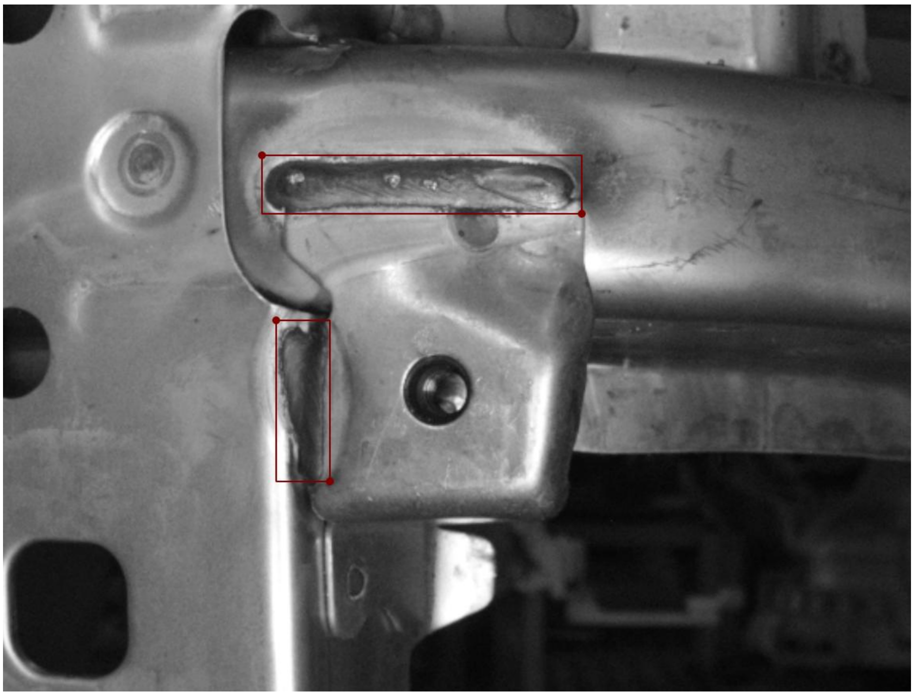


Figure 19 – Original sample image from the welding process, captured with a low-resolution camera (`weld-low` dataset). The bounding boxes highlights two weld bead regions used in this research.



acquired in a higher resolution (as detailed in Figure 21b), because it was observed that many defects were concentrated over this bead due to the difficulty of positioning the robot during the welding process.

Furthermore, as part of the proposed datasets, we also provided a ground truth file for each image — refer to Figure 21c. Ground truth entries correspond to an axis-aligned rectangular box, manually annotated, and associated with one of the four types of weld defects addressed in this study (pores, deposits, discontinuities, and stains). A region of interest is also delimited by an axis-aligned rectangle that encloses each weld bead.

As detailed earlier, this project focus in four types of weld defects, as illustrated in

Figure 20 – Original sample image from the welding process, captured with a high-resolution camera (`weld-high` dataset). The bounding box highlight a horizontal weld bead region, which is more prone to defects.

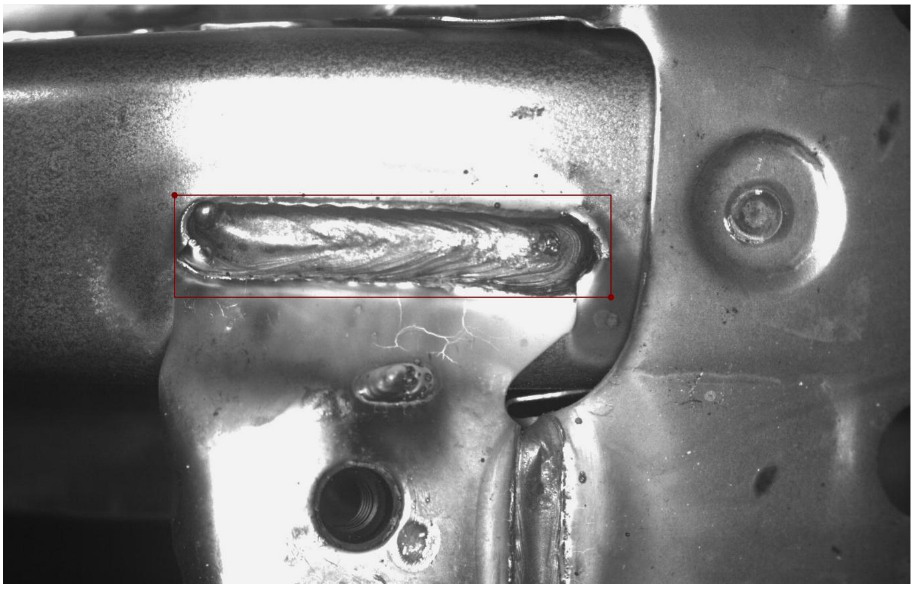
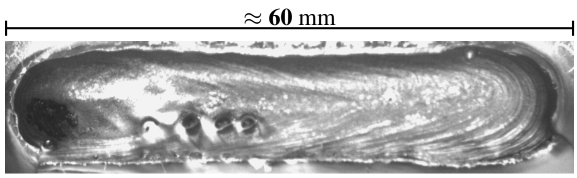
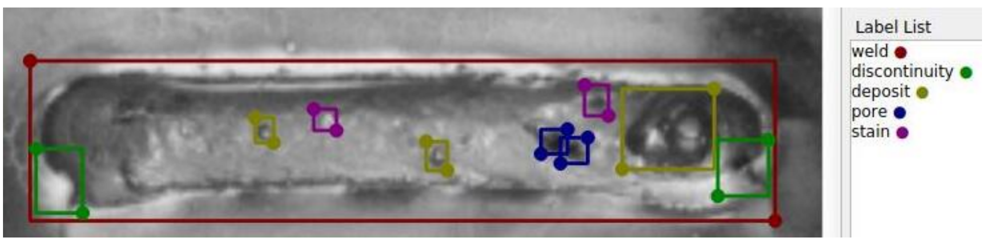


Figure 21 – Image samples and annotations example: (a) images of two distinct weld bead regions from the `weld-low` dataset, with corresponding physical dimensions; (b) image from the `weld-high` dataset with corresponding physical dimension; and (c) manual annotations of the four types of defects and the manual delimitation of the weld bead region. For display purposes, the images were independently re-scaled. Source: Author.

(a) `weld-low` images (avg. width 155 pixels; avg. height 40 pixels).



(b) `weld-high` image (avg. width 779 pixels; avg. height 192 pixels).



(c) Manual annotations.

Figure 21, namely:

- ***pore***: small filling gaps, that occur due to air bubbles in the process of melting the weld material;
- ***deposit***: an excess of welding material along the weld bead, resulting in an uneven surface;
- ***discontinuity***: a partial absence of weld material along the bead, forming regions with a lack of expected thickness;
- ***stain***: a slight alteration in the surface color of the weld, that may indicate underlying defects within the weld bead.

It is worth noting that the purpose of investigating the `weld-low` dataset is to reduce automation costs since we can use a reduced network bandwidth to transmit the images (in a robotic welding line, the network links can be limited) and decreased computing power for image processing, given that high-resolution images contain more than ten times the number of pixels compared to low-resolution images. Despite guaranteeing a millimeter positioning accuracy of the parts to be joined, the robotic welding process still experiences some variation during the melting of the weld material (which forms the bead). As a result, variations in the size of the weld beads, including both width and length, can be observed.

Table 5 and in Figure 22 summarize the information regarding both datasets.

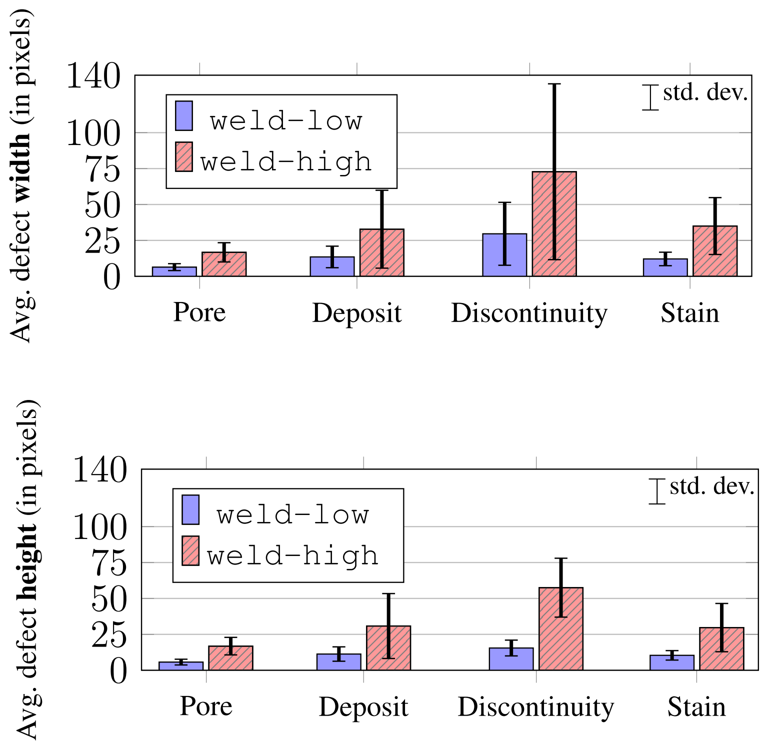
Table 5 – Statistics for `weld-low` and `weld-high` welding datasets: the columns indicate the total number of weld bead images and the number of welding defects of each category identified in the manual annotation process. Source: Author.

	#Weld beads	#Pore	#Deposit	#Disc.	#Stain
`weld-low`	2,000	3,427	1,141	4,243	4,126
`weld-high`	1,022	553	1,379	2,798	2,886
Total	3,022	3,961	2,442	6,607	6,321

5.2 ARCHITECTURE BASELINE

The architecture scheme used for weld defect recognition is shown in Figure 23. The input are weld bead images, not necessarily with the same dimensions. Each image is resized to a square shape with a fixed size — this size is not changed during training and testing — since the network restricts the input to square images. For this purpose, a *letterbox* resizing is applied to scale the original image while maintaining the aspect ratio; more specifically, as the weld

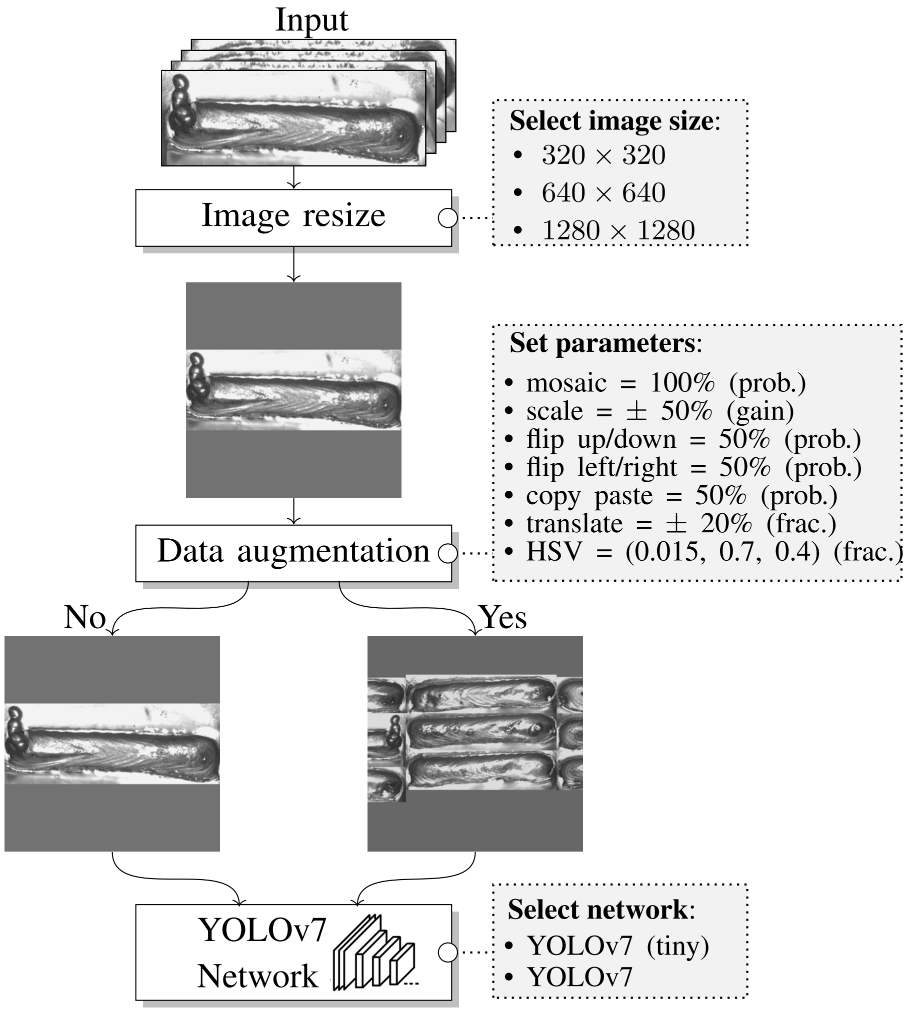
Figure 22 – Defect size histogram according to average width and height (in pixels). Source: Author.



regions are horizontally oriented, their long side is scaled to the selected size while the resized shorter side is padded with a gray value. We explored three image input sizes: the 320×320 pixels resolution, to evaluate whether scaling up the low-resolution images (average width of 155 pixels as shown in Figure 21a) to this closer resolution is beneficial or not; the 640×640 resolution, which is a standard dimension widely used in the literature for the adopted deep model (WANG *et al.*, 2023); and the 1280×1280 resolution, also a standard size, and important to evaluate if a high resolution `weld-high` image (average width of 779 pixels as shown in Figure 21b) are adversely affected by upscaling or downscaling for defect recognition.

Data augmentation artificially increases the training set by creating modified copies of the image samples to improve model generalization (reduce overfitting). For our problem, a classifier that is invariant to position, rotation, scale, and lighting changes is important. Without a controlled environment (closed chamber), it may be difficult to achieve absolute illumination control on a production line, especially in robotic welding processes. Also, depending on the region being inspected, the resolution and position of weld beads can significantly vary, as the camera acquisition setup needs to be adjusted to the robot's working space. Such changes can be noted by comparing the `weld-low` and `weld-high` image samples. Therefore, we extensively experimented with image transformations to evaluate the impact of data augmentation

Figure 23 – Architecture scheme for weld defect recognition. Source: Author.



on the architecture. The best parameters we found are shown in Figure 23, namely:

- mosaic: involves combining some images into a single image to increase the diversity of training data. This parameter controls the likelihood of applying mosaic augmentation during training, e.g. 100% means that this augmentation will be applied to every training batch.

- scale: random resizing of images, allowing the model to be more robust to variations in object size. This parameter random adjusts the scale of the images by a factor within the interval range +50% (zoom-in) and -50% (zoom-out).

- flips: these parameters define the probability of flipping images vertically (up and down) or horizontally (left and right) during the training.

- copy paste: involves the probability of pasting objects from one image onto another, helping to increase the variety of object appearances and backgrounds.
- translate: defines the maximum fraction of translation. It is used for randomly shifting the images horizontally or vertically.
- HSV: this parameter specifies the range for Hue, Saturation, and Value adjustments.

For the detection/classification problem, we used the YOLOv7 (WANG *et al.*, 2023) network, from 2022, a robust deep architecture that surpassed all known object detectors/classifiers in both speed and accuracy for real-time applications, according to the authors. The YOLO (You Only Look Once) model directly predicts bounding boxes and class probabilities for objects in an image using an end-to-end model (REDMON *et al.*, 2016), unlike other famous object detectors, such as R-CNN and Fast R-CNN, which are multiple-stage methods. YOLO divides the input image into a grid and predicts objects within cells. For each cell, the model samples multiple bounding boxes, defined by coordinates and a confidence score indicating the likelihood of a box to contain an object. Additionally, class probabilities are assigned to each bounding box (regression task) to determine the object's class (classification task). The final output is a set of bounding boxes and associated class probabilities that collectively represent the detected objects in the image.

Over time, YOLO has undergone numerous modifications to improve performance and real-time operation for different platforms. Notably, the YOLOv7 model, used in our work, presents different contributions (WANG *et al.*, 2023): (i) it proposes several trainable bag-of-freebies methods (i.e., a combination of techniques such as data augmentation, learning rate warmup, optimized anchor boxes, among others) so that real-time object detection can significantly improve its accuracy without increasing the inference cost; (ii) model re-parameterization; (iii) a modification of the dynamic label assignment strategy for different output layers; and (iv) parameter reduction. YOLOv7 additionally uses CSPDarknet (BOCHKOVSKIY *et al.*, 2020) as its backbone network and a head composed of Path Aggregation Network (PANet) modules (LI *et al.*, 2023). This helps integrate features from different network scales, enhancing the ability of the model to detect objects of various sizes. It also optimizes inference by employing the Complete Intersection over Union (CIOU) loss (WANG; SONG, 2021) to bounding box prediction. The model incorporates anchor-free object detection to eliminate the need for anchor box selection.

Moreover, it includes techniques like the Spatial Attention Module (SAM) (WOO *et al.*, 2018) to focus on important spatial locations.

Finally, the YOLOv7 authors designed models for edge GPU, normal GPU, and cloud GPU, respectively, YOLOv7-tiny (6.2 million of parameters to optimize), YOLOv7 (36.9 million), and YOLOv7-W6 (70.4 million), as well as other variations. We considered the YOLOv7-tiny and YOLOv7 models since speed is crucial in industry.

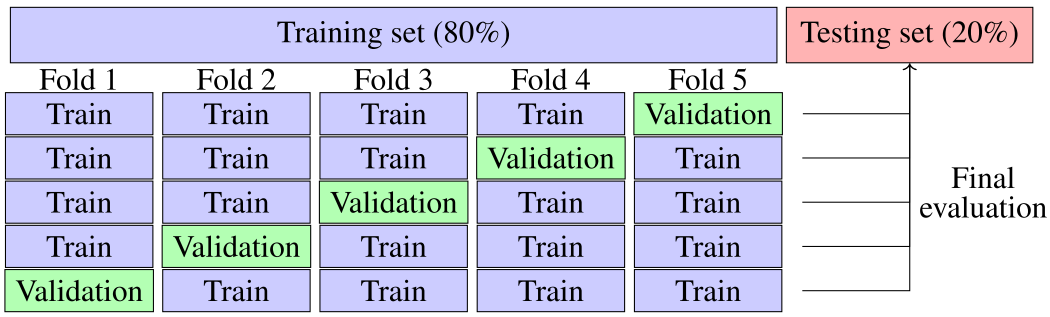
6 EXPERIMENTAL RESULTS FOR WELDING DEFECTS

In this chapter, we describe the test settings, evaluation metrics, and cross-validation methods used, along with the results obtained for detecting and classifying welding defects in the proposed dataset. Additionally, we provide an analysis of each defect class and discuss the impact of image resolution on the performance.

6.1 SETTINGS

In our experiments, both `weld-low` and `weld-high` datasets were partitioned into 80% of the image samples for training and 20% for testing (holdout subset). For training, we used a 5-fold cross-validation technique, as shown in Figure 24, saving the best model in 50 epochs (based on the performance over the validation fold) to be evaluated in the end with the testing set. The hardware setup is an Intel i7-10700 2.9GHz, 128GB of RAM, and a GPU NVIDIA RTX 3090 (24 GB).

Figure 24 – Training/testing setup used in our experiments. Source: Author.



6.2 METRICS

The architecture performance is reported according to well-known metrics, such as precision (P), recall (R), and F-score (F) — the harmonic mean of precision and recall, that is, $F = 2/(1/P + 1/R)$. A detected region with an overlap, according to the intersection over union (IoU) relation, of at least 50% with a ground truth region is considered a true positive; otherwise, it is a false positive. A region is also a false positive when the class assigned differs from the ground truth. Undetected ground truth regions are considered false negatives. We also report the Mean Average Precision (mAP) metric for an IoU of 50%, as defined in the PASCAL

VOC challenge, since it is widely used by other benchmark challenges, such as ImageNET and Google Open Image Challenge, and it is a standard metric to evaluate object detection models. All metrics are weighted according to the classification confidence score.

6.3 RESULTS

In this section the results in classification will be discussed, regarding the impact of image size, performance per class, models size, cross validation and defect granularity impact on performance.

6.3.1 Impact of image size and data augmentation

Table 6 summarizes the architecture results shown in Figure 23 with a YOLOv7-tiny network, by varying image sizes and with/without data augmentation. As can be seen, regardless of the image size and dataset resolution, the data augmentation tends to effectively improve the performance for all evaluating metrics — an exception occurred with the `weld-high` dataset and images of 320×320 pixels, which may mostly be due to losing details in downsampling. The improvement shows the importance a greater diversity of the augmented data has during the training step.

Table 6 – Architecture performance for `weld-low` and `weld-high` datasets using a YOLOv7-tiny network, varying the image size, and with/without data augmentation. The results are an average over the testing dataset for the best models found during the optimization using the 5-fold technique. The maximum standard deviation is 0.06 for experiments with an image resolution of 320×320 pixels, and 0.01 for experiments with an image resolution of 640×640 or 1280×1280 pixels. The boldfaced values indicate the best results for each dataset. Source: Author.

Dataset	Img. Size	Data aug.	P	R	F	mAP
`weld-low`	320×320	✗	0.49	0.46	0.47	0.39
	320×320	✓	0.57	0.57	0.57	0.54
	640×640	✗	0.57	0.54	0.56	0.49
	640 × 640	✓	0.66	**0.62**	**0.64**	**0.64**
	1280×1280	✗	0.59	0.54	0.56	0.52
	1280×1280	✓	**0.67**	0.61	0.63	**0.64**
`weld-high`	320×320	✗	0.44	0.42	0.43	0.36
	320×320	✓	0.51	0.39	0.43	0.35
	640×640	✗	0.65	0.53	0.58	0.54
	640×640	✓	0.67	0.62	0.64	0.64
	1280×1280	✗	0.66	0.59	0.62	0.60
	1280 × 1280	✓	**0.72**	**0.64**	**0.67**	**0.69**

In the `weld-low` dataset, a weld bead region has an average width of 115 pixels —

refer to Figure 21a. Considering the results using data augmentation and the image size variations, one can see that an image resolution of 640×640 pixels achieved a meaningful increase in metrics compared to a 320×320 resolution. Images of 320×320 pixels lead to F-score and mAP results of 0.57 and 0.54, respectively, while images of 640×640 lead to F-score and mAP results of 0.64 and 0.64. Therefore, there were gains of 7 and 10 percentage points. However, the performance kept relatively stable for images of 1280×1280 pixels.

Regarding the `weld-high` dataset, the performance improved along with the increasing image sizes, reaching the best results with an image resolution of 1280×1280 pixels. It is worth noting that scaling down the weld bead images below the original resolution size has a major negative impact on performance, suggesting that the reduction of images can lead to the loss of important details of already challenging small features in the images. As shown in Figure 21, a high-resolution weld bead region has an average of 779 pixels width. Considering the results with data augmentation for the `weld-high` dataset, from 320×320 to 640×640 resolution, there were gains of 21 percentage points in F-score (0.43 against 0.64, respectively) and 29 percentage points in mAP (0.35 against 0.64). There are also percentage point gains — 3% in F-score and 5% in mAP — comparing resolution 640×640 against 1280×1280.

From this point on, the remaining experiments will concern the best architectures shown in Table 6, namely a 640×640 image resolution with data augmentation for `weld-low` dataset (best F-score and mAP), and 1280×1280 image resolution with data augmentation for `weld-high` dataset (best F-score and mAP).

6.3.2 Performance analysis per class

Figure 25 shows the confusion matrix for the `weld-low` dataset. There are not many misclassifications regarding pores and other defects, as well as, discontinuities and other defects. Misclassifications between stains and deposits are most likely to occur — as some light-colored stains can resemble deposits. In general, false positives and false negatives are the primary types of misclassifications for each class, as the classes have similarities with the background (BG). Figure 26 show manual annotations, as well as correct and incorrect region classifications for sample images from `weld-low` and `weld-high` sets.

The confusion matrix for the `weld-high` dataset is shown in Figure 27. The pore class is rare in the `weld-high` dataset, comprising only 5.5% as indicated in Table 5, which may explain the decline in the classification results, compared to the `weld-low` dataset, especially

Figure 25 – Confusion matrix for `weld-low` dataset with a YOLOv7-tiny network, image resolution of 640×640 pixels, and data augmentation. The results are an average of 5-fold over the testing set. Source: Author.

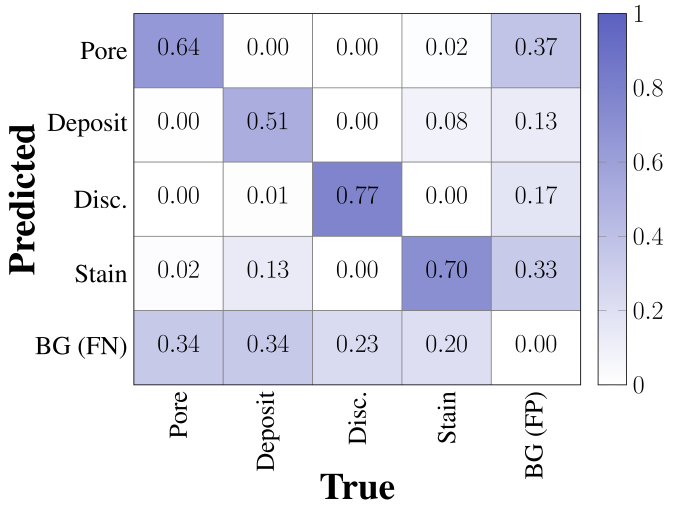
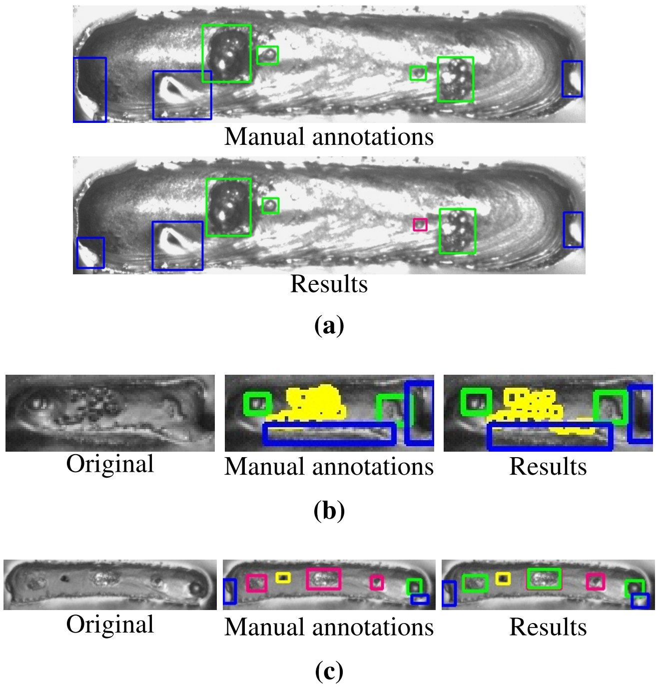
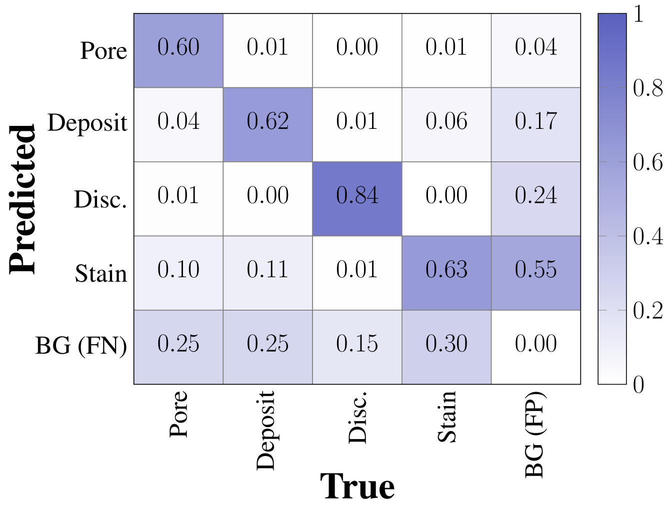


Figure 26 – Detection and classification results for `weld-high` (a) and `weld-low` (b,c) sample images with a tiny network. Yellow regions represent porous defects, blue regions represent discontinuities, green regions represent deposits, and pink regions represent stains. For `weld-low` images, we also show the original image for better visualization of the results. Source: Author.



the confusion with the stain class. As in the `weld-low` dataset, the false positives and negatives account for most misclassifications. The number of false positives was greatly reduced for the pore class; even though there are few samples for the class, this can be an effect of the higher dimension of pores compared to the `weld-low` dataset, so that the model can capture better features to distinguish pores and background. There was an increase in true positives for the deposit class, from 0.51 (`weld-low`) to 0.62 (`weld-high`). This is also due to the larger size of the deposits contained in the `weld-high` dataset in comparison to `weld-low`, where deposits are smaller. Misclassification (false positives and false negatives) between the stain class and the background had increased — that can be a result of the reduction of samples for `weld-high` combined with greater detail of the irregularities on the surface of the weld bead, which can be mistaken for stains.

Figure 27 – Confusion matrix for `weld-high` dataset using an architecture with a YOLOv7-tiny network, image resolution of 1280×1280 pixels, and data augmentation. The results are an average of 5-fold over the testing set. Source: Author.



6.3.3 Small versus larger models

Table 7 shows the performance comparison between a low-cost YOLOv7-tiny network, with only 6.2 million parameters to fine-tune, against a larger model, YOLOv7, with 36.9 million parameters. As can be seen, a more complex model for our problem did not bring a relevant gain. Furthermore, as explored by Nguyen *et al.* (2022), the tiny model is much faster for inference in low-cost edge GPU devices, such as those used in industrial processes. For instance, the authors reported 40 FPS for inference with a YOLOv7-tiny model, against 17 FPS with a YOLOv7 model, on an NVIDIA Jetson AGX Xavier; and 16 FPS for inference with a YOLOv7-tiny model, against 3 FPS with a YOLOv7 model, on an NVIDIA Jetson Nano (both results for a 640×640 image resolution).

Table 7 – Performance comparison between architectures with YOLOv7-tiny and YOLOv7 for `weld-low` and `weld-high` datasets. For all values, the maximum standard deviation is below or equal 0.01. Source: Author.

Dataset	Network	Img. size	Data aug.	P	R	F	mAP
`weld-low`	YOLOv7-tiny	640 × 640	✓	**0.66**	0.62	**0.64**	**0.64**
	YOLOv7	640 × 640	✓	0.65	**0.63**	**0.64**	0.63
`weld-high`	YOLOv7-tiny	1280 × 1280	✓	0.72	**0.64**	**0.67**	0.69
	YOLOv7	1280 × 1280	✓	**0.73**	0.63	**0.67**	**0.70**

6.3.4 Cross-validation and dataset fusion

Figure 28(a) shows an experimental setup that assesses the architecture's ability to generalize across different image resolution size sets, measuring its robustness and reliability beyond the initially used training data. For this purpose, we trained on image samples from `weld-low` dataset and tested over `weld-high` samples (and vice-versa). As reported in rows Cross_1 and Cross_2 of Table 8, the best architectures discussed in Section 6.3.1 seem to be highly dependent on the scale of features learned, as we can note from the poor performance results. Figure 28(b) shows an experimental setup with the `weld-low` and `weld-high` datasets merged into a single set to assess the architecture's ability to learn from combined data of multiple sources. As seen in Table 8, the Fusion_1 experiment shows that the trained model maintained similar results to the model trained specifically with the `weld-low` dataset and performed slightly worse when compared to the model trained with the `weld-high` dataset. Moreover, the Fusion_2 and Fusion_3 experiments show that training with merged samples did not compromise performance for any particular set, whether from `weld-low` and `weld-high` samples. In general, the results from fusion experiments indicate that although the architecture dealt with differences in image resolution between the datasets `weld-low` and `weld-high` , the increase of training samples did not bring a performance gain.

6.3.5 Coarse versus fine-grained defect classification

In another round of experiments, we grouped the four categories of welding defects in our dataset (pore, deposit, discontinuity, and stain) into a broad category referred to here as defect — that is, the original multiclass classification problem was modified into a more straightforward (coarse) binary classification problem. In machine learning, coarse classification involves grouping objects into broad categories based on general features or characteristics, often sacrificing specificity for simplicity and efficiency. In contrast, fine classification involves a more

Figure 28 – Setup for cross-validation (a) and dataset fusion (b) experiments regarding `weld-low` and `weld-high` datasets. Source: Author.

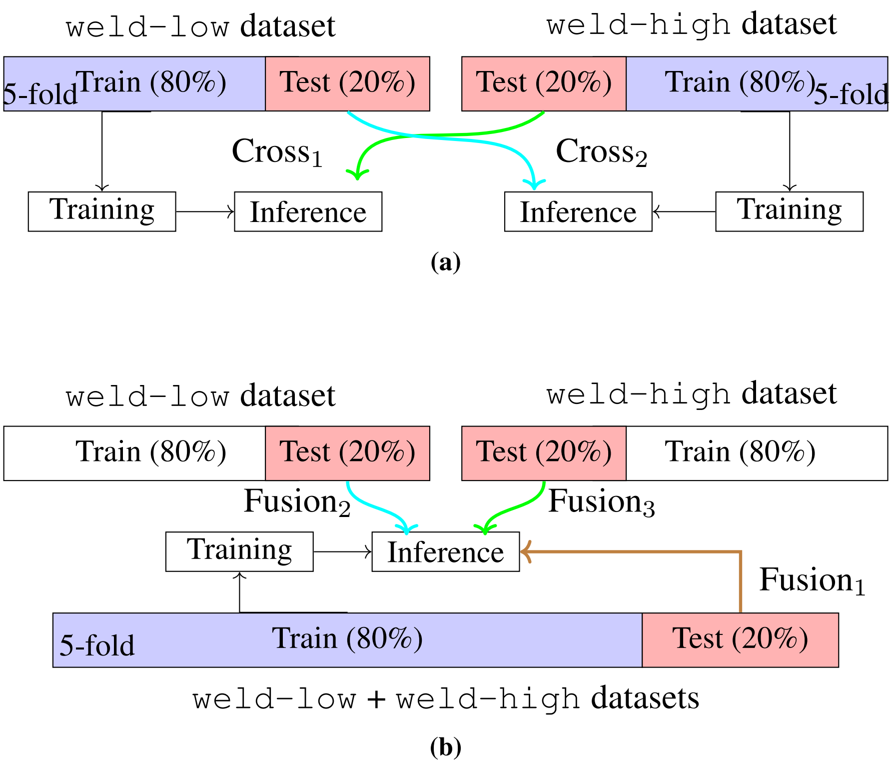


Table 8 – Performance regarding cross-validation and dataset fusion experiments, for architecture shown in Figure 23, by using a YOLOv7-tiny network. For these experiments, the same image size was used for training and inference. For all values, the maximum standard deviation is below or equal 0.02. Source: Author.

	Img. Size	Data aug.	P	R	F	mAP
Cross$_1$	640 × 640	✓	0.32	0.35	0.33	0.25
	1280 × 1280	✓	**0.35**	**0.39**	**0.37**	**0.29**
Cross$_2$	640 × 640	✓	**0.30**	**0.26**	**0.28**	**0.19**
	1280 × 1280	✓	0.28	0.25	0.26	0.18
Fusion$_1$	640 × 640	✓	**0.68**	0.62	0.64	0.65
	1280 × 1280	✓	0.66	**0.65**	**0.65**	**0.67**
Fusion$_2$	640 × 640	✓	**0.66**	0.63	**0.64**	**0.64**
	1280 × 1280	✓	0.63	**0.64**	0.63	**0.64**
Fusion$_3$	640 × 640	✓	0.69	0.61	0.65	0.65
	1280 × 1280	✓	**0.70**	**0.65**	**0.67**	**0.69**

detailed analysis, where objects are categorized by using a higher level of granularity (ZHENG *et al.*, 2020).

The goal is to evaluate the overall architecture baseline performance concerning the presence or absence of welding defects rather than a granular differentiation between categories of defects. As shown in Table 9, the F-score and mAP for `weld-low` dataset increased by 8 and 11 percent points, respectively. In contrast, the F-score and mAP for `weld-high` dataset increased 7 and 8 percent points, respectively, showing that many errors are not due to non-detections but assigning the defect to the incorrect category. In this way, a hybrid method that first applies coarse classification to group objects into broad categories before refining the classification at finer levels of detail can be a promising source of future research.

Table 9 – Performance comparison between architectures trained for a multiclass (fine) classification problem against a binary (coarse) classification problem. For all values, the maximum standard deviation is below or equal 0.01. Source: Author.

Dataset	Network	Classification	Img. size	Data aug.	P	R	F	mAP
`weld-low`	YOLOv7-tiny	Fine	640×640	✓	0.66	0.62	0.64	0.64
	YOLOv7-tiny	Coarse	640×640	✓	0.74	0.70	0.72	0.75
`weld-high`	YOLOv7-tiny	Fine	1280×1280	✓	0.72	0.64	0.67	0.69
	YOLOv7-tiny	Coarse	1280×1280	✓	0.77	0.72	0.74	0.77

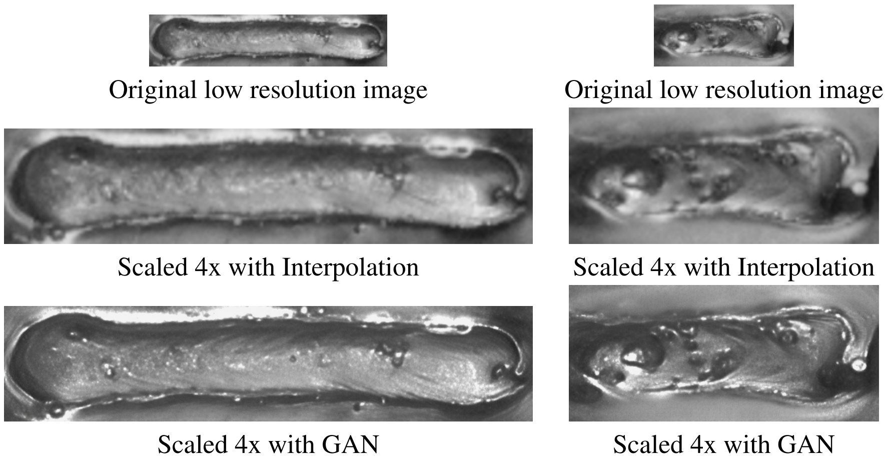
6.4 DISCUSSION

As demonstrated so far, more comprehensive and properly annotated public databases are needed for real weld defect detection problems. In this sense, the efforts of this work, in addition to providing the dataset, promote a comparison baseline for future analyses. We believe that there are numerous open challenges for this study, such as:

- the proposition of an ensemble or architecture dedicated to the problem in question, aiming to increase general detection performance;
- using the proposed dataset in conjunction with generative models, e.g., GAN, to create more extensive databases and data augmentation procedures;
- exploring knowledge transfer between networks in more detail, for example, using `weld-low` or `weld-high` as a starting point for another network or model and avoiding the need for a large set of samples for initial training;
- inspecting models like Visual Transformers (DOSOVITSKIY *et al.*, 2021) that may be promising if the probability increases when a defect appears, e.g., pores are usually associated with some other type of defect;
- performing semantic annotation (with natural language) of defects instead of labels to allow a more detailed description and closer to a practical application for line operators;

- augmenting the fine-grained annotations of the dataset by subdivision of classes, e.g., partition the deposit class into one covering larger and unique deposits and another covering small and scattered deposits along the weld; or subdivide the discontinuities into those found at the edge of the bead and those positioned within the bead;
- further annotating the datasets by other specialists to cover more subtle surface defects, generally relevant in welds that are exposed in the final product;
- comparing other low-cost architectures aimed at detecting and classifying objects in embedded systems, aiming to develop an edge device;
- exploring the proposed dataset with transfer learning from other welding datasets (KUMARESAN *et al.*, 2021b; PAN *et al.*, 2020).
- exploring super-resolution deep methods (WANG *et al.*, 2021), as shown in Figure 29, to improve the quality of low-resolution images in order to strike a balance between the speed in acquiring images during a line production and a higher performance for high-resolution images.

Figure 29 – Image scaling with traditional interpolation methods against a Generative Adversarial Network (GAN) super-resolution method: top row, two original image samples from `weld-low` dataset; middle row, the original `weld-low` samples were scaled four times the original resolution size with a bilinear interpolation algorithm; and, bottom row, the original `weld-low` samples were scaled four times with ESRGAN (WANG *et al.*, 2021) — a Generative Adversarial Network (GAN) for image super-resolution — trained only with `weld-high` samples. Source: Author.



7 CONCLUSIONS

In this thesis, we addressed the detection of defects on metallic surfaces at two key stages in the automotive production line, specifically stamping and welding. Stamping defects, which may appear on exposed vehicle parts, need to be identified promptly to prevent the need for costly repairs or replacement of parts. Detecting welding defects is equally important, as weld beads form the structural bonds between vehicle components and are directly related to the safety and integrity of the vehicle. Ensuring the early and accurate identification of these defects is essential for maintaining both production efficiency and safety standards.

Regarding stamping defects, we developed a framework to detect mild and severe imprint defects on metallic surfaces. The proposed method integrates detection and classification, using a convolutional neural network, combined with tracking across video frames. This integration is important because applying a detector independently to each frame ignores the temporal coherence between adjacent frames, leading to uncorrelated detections for the same defect. By exploring temporal redundancy, the framework identifies missing detections that likely correspond to the same defect, evaluates detection consistency by tracking the number of occurrences of a defect along its path, and assesses classification consistency based on the assigned labels throughout the tracked sequence. In our experiments, the proposed framework achieved a mean average precision of 76.21% to detect and classify mild and severe defects. For severe imprints, the framework obtained precision and recall values of 90% and 92%, respectively, when discarding sequences with only one detection. These are promising results considering that severe defects are critical and need quick identification, since they may cause the disposal of the stamped parts. Although the results for mild defects were slightly lower, with precision at 69% and recall at 85%, this is not critical as they can often be fixed along the production process.

Despite the progress made in detecting imprint defects, several challenges remain. These defects are particularly difficult to detect through visual inspection due to their small size and the influence of light incidence and reflection on the stamped parts. The limited availability of defect images from production lines further complicates the process of building large training datasets. As a result, there are still significant opportunities to enhance the detection of such defects. To mitigate the issue of limited data, approaches such as transfer learning and synthetic dataset generation using Generative Adversarial Networks (GANs) can be explored to increase the sample size for training. Furthermore, leveraging specialized neural architectures designed for small object detection may improve the accuracy in identifying small-scale defects. Furthermore, methods like Simple Online and Real-time Tracking (SORT) can accelerate the tracking process by operating directly over deep features, while structured light techniques can enhance the visibility of imprints by highlighting their surface characteristics. These strategies may lead to substantial improvements in the detection and classification of such defects.

Concerning welding defects, we investigated the impact of various image resolutions, data augmentation techniques, and architectural complexities, using the YOLOv7 model as a baseline, to detect and classify defects such as pores, deposits, discontinuities, and stains on weld bead surfaces. We also introduced the LoHi-WELD dataset, a novel and publicly available dataset, spanning more than 22,000 images of welding defects captured from an industrial production line. This dataset adds significant complexity and variability to the problem. Our study revealed that the performance of a smaller architecture (YOLOv7-tiny) against a larger model (YOLOv7) is favorable in an industrial production line, as it suggests that inexpensive hardware (edge devices) can be used for the task. YOLOv7-tiny obtained mean average precision (mAP) of 64% on the low-resolution dataset (`weld-low`) and 69% on the high-resolution dataset (`weld-high`), a 5% gain from `weld-low` to `weld-high` with nearly half the number of weld beads (1,022 high-resolution weld beads against 2,000 low-resolution weld beads). These results highlight the potential of using cost-effective edge devices for defect detection. Cross-dataset and fusion generalization experiments were also conducted, demonstrating that performance can generally be maintained in a fusion scenario. Additionally, the detection of welding defects and subsequent binary classification (defect/non-defect) achieved 77% of mean average precision (mAP) when considering the high-resolution dataset and 75% of mAP when considering the low-resolution dataset. These findings suggest that a coarser classification, with defects grouped into broader categories, followed by a finer classification within groups can be a future research direction.

Key areas for further exploration in weld defect recognition include developing specialized models to improve detection performance, using generative models to enhance data augmentation, and leveraging knowledge transfer between networks to reduce the need for extensive initial training. Additionally, investigating advanced models, employing semantic annotations for practical defect descriptions, and enhancing dataset annotations with more detailed classifications could also provide significant benefits. Finally, utilizing super-resolution methods to balance image quality with acquisition speed presents another promising avenue for improving defect detection.

In summary, this thesis presented a comprehensive study of two critical stages in vehicle manufacturing, introducing two substantial and novel datasets of industrial defects and proposing methods for their detection and classification. This work demonstrates the practical potential of computer vision and deep learning systems to enhance the quality and efficiency of industrial production processes. The findings highlight the potential for continued advancements in industrial inspection through the use of innovative datasets and methodologies.

REFERENCES

ANITHA, DB; RAO, Mahesh. A survey on defect detection in bare pcb and assembled pcb using image processing techniques. *In*: IEEE. **2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)**. *[S.l.]*, 2017. p. 39–43.

BLOCK, Sylvio Alexandre Biasuz. Mestrado em Computação Aplicada CPGGA (UTFPR), **Inspeção e classificação de picos em peças estampadas de metal utilizando rede neural convolucional**. 2018. 67 p.

BLOCK, Sylvio B; SILVA, Ricardo D Da; LAZZARETTI, Andre E; MINETTO, Rodrigo. Lohi-weld: a novel industrial dataset for weld defect detection and classification, a deep learning study, and future perspectives. **IEEE Access**, IEEE, 2024.

BLOCK, Sylvio Biasuz; SILVA, Ricardo Dutra da; DORINI, Leyza Baldo; MINETTO, Rodrigo. Inspection of imprint defects in stamped metal surfaces using deep learning and tracking. **IEEE Transactions on Industrial Electronics**, v. 68, n. 5, p. 4498–4507, 2021.

BOCHKOVSKIY, Alexey; WANG, Chien-Yao; LIAO, Hong-Yuan Mark. Yolov4: Optimal speed and accuracy of object detection. **arXiv preprint arXiv:2004.10934**, 2020.

BOLME, D. S.; BEVERIDGE, J. R.; DRAPER, B. A.; LUI, Y. M. Visual object tracking using adaptive correlation filters. *In*: **IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. *[S.l.: s.n.]*, 2010. p. 2544–2550. ISSN 1063-6919.

CAI, N.; CEN, G.; WU, J.; LI, F.; WANG, H.; CHEN, X. Smt solder joint inspection via a novel cascaded convolutional neural network. **IEEE Transactions on Components, Packaging and Manufacturing Technology**, v. 8, n. 4, p. 670–677, April 2018. ISSN 2156-3950.

CHANDOLA, Varun; BANERJEE, Arindam; KUMAR, Vipin. Anomaly detection: A survey. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 41, n. 3, jul 2009. ISSN 0360-0300. Available at: https://doi.org/10.1145/1541880.1541882.

CHANG, S. J.; PARK, J. B. Wire mismatch detection using a convolutional neural network and fault localization based on time–frequency-domain reflectometry. **IEEE Transactions on Industrial Electronics**, v. 66, n. 3, p. 2102–2110, March 2019. ISSN 0278-0046.

CHANG, Yasheng; WANG, Weiku. A deep learning-based weld defect classification method using radiographic images with a cylindrical projection. **IEEE Trans. on Inst. and Measurement**, v. 70, p. 1–11, 2021.

CHEN, F.; JAHANSHAHI, M. R. Nb-cnn: Deep learning-based crack detection using convolutional neural network and naïve bayes data fusion. **IEEE Transactions on Industrial Electronics**, v. 65, n. 5, p. 4392–4400, May 2018. ISSN 0278-0046.

DENG, Honggui; CHENG, Yu; FENG, Yuxin; XIANG, Junjiang. Industrial laser welding defect detection and image defect recognition based on deep learning model developed. **Symmetry**, MDPI, v. 13, n. 9, p. 1731, 2021.

DOSOVITSKIY, Alexey; BEYER, Lucas; KOLESNIKOV, Alexander; WEISSENBORN, Dirk; ZHAI, Xiaohua; UNTERTHINER, Thomas; DEHGHANI, Mostafa; MINDERER, Matthias; HEIGOLD, Georg; GELLY, Sylvain; USZKOREIT, Jakob; HOULSBY, Neil. An image is worth 16x16 words: Transformers for image recognition at scale. **International Conference on Learning Representations (ICLR)**, 2021. Available at: https://arxiv.org/abs/2010.11929.

DU, Yunhao; ZHAO, Zhicheng; SONG, Yang; ZHAO, Yanyun; SU, Fei; GONG, Tao; MENG, Hongying. Strongsort: Make deepsort great again. **IEEE Transactions on Multimedia**, IEEE, v. 25, p. 8725–8737, 2023.

Everingham, Mark; Gool, Luc Van; Williams, Christopher K. I.; Winn, John; Zisserman, Andrew. The pascal visual object classes (voc) challenge. **International Journal of Computer Vision**, v. 88, n. 2, p. 303–338, Jun. 2010.

GAO, Ying; LIN, Jiqiang; XIE, Jie; NING, Zhaolong. A real-time defect detection method for digital signal processing of industrial inspection applications. **IEEE Transactions on Industrial Informatics**, IEEE, v. 17, n. 5, p. 3450–3459, 2020.

GUO, Runyuan; LIU, Han; XIE, Guo; ZHANG, Youmin. Weld defect detection from imbalanced radiographic images based on contrast enhancement conditional generative adversarial network and transfer learning. **IEEE Sensors Journal**, IEEE, v. 21, n. 9, p. 10844–10853, 2021.

HANBAY, Kazım; TALU, Muhammed Fatih; ÖZGÜVEN, Ömer Faruk. Fabric defect detection systems and methods—a systematic literature review. **Optik**, Elsevier, v. 127, n. 24, p. 11960–11973, 2016.

HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. Deep residual learning for image recognition. *In*: **IEEE Conference on Computer Vision and Pattern Recognition**. *[S.l.: s.n.]*, 2016.

HERRERO, F. Javier de la Calle; GARCíA, D. F.; USAMENTIAGA, R. Inspection system for rail surfaces using differential images. **IEEE Transactions on Industry Applications**, v. 54, n. 5, p. 4948–4957, Sep. 2018. ISSN 0093-9994.

HONG, Yuxiang; YANG, Mingxuan; CHANG, Baohua; DU, Dong. Filter-PCA-Based process monitoring and defect identification during climbing helium arc welding process using de-svm. **IEEE Transactions on Industrial Electronics**, v. 70, n. 7, p. 7353–7362, 2023.

HONG, Yuxiang; YANG, Mingxuan; JIANG, Yuxuan; DU, Dong; CHANG, Baohua. Real-time quality monitoring of ultrathin sheets edge welding based on microvision sensing and socifs-svm. **IEEE Transactions on Industrial Informatics**, v. 19, n. 4, p. 5506–5516, 2023.

IGLESIAS, Carla; MARTÍNEZ, Javier; TABOADA, Javier. Automated vision system for quality inspection of slate slabs. **Computers in Industry**, Elsevier, v. 99, p. 119–129, 2018.

Kammel, S.; Puente Leon, F. Deflectometric measurement of specular surfaces. **IEEE Transactions on Instrumentation and Measurement**, v. 57, n. 4, p. 763–769, April 2008. ISSN 0018-9456.

KNAUER, Markus C.; KAMINSKI, Jurgen; HAUSLER, Gerd. Phase measuring deflectometry: a new approach to measure specular free-form surfaces. *In*: . *[S.l.: s.n.]*, 2004. v. 5457, p. 366–376.

KONG, H.; YANG, J.; CHEN, Z. Accurate and efficient inspection of speckle and scratch defects on surfaces of planar products. **IEEE Transactions on Industrial Informatics**, v. 13, n. 4, p. 1855–1865, Aug 2017. ISSN 1551-3203.

KUMARESAN, Samuel; AULTRIN, K. S. Jai; KUMAR, S. S.; ANAND, M. Dev. Transfer learning with CNN for classification of weld defect. **IEEE Access**, v. 9, p. 95097–95108, 2021.

KUMARESAN, Samuel; AULTRIN, K. S. Jai; KUMAR, S. S.; ANAND, M. Dev. Transfer learning with cnn for classification of weld defect. **IEEE Access**, v. 9, p. 95097–95108, 2021.

LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. **Nature**, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 5 2015. ISSN 0028-0836.

LEON, F. Puente; KAMMEL, S. Inspection of specular and painted surfaces with centralized fusion techniques. **Measurement**, v. 39, n. 6, p. 536 – 546, 2006. ISSN 0263-2241.

LI, Rui; ZHANG, Baopeng; LIU, Wei; TENG, Zhu; FAN, Jianping. Panet: An end-to-end network based on relative motion for online multi-object tracking. **ACM Transactions on Multimedia Computing, Communications and Applications**, ACM New York, NY, v. 19, n. 6, p. 1–21, 2023.

LI, Xiaoli; TSO, Shiu Kit; GUAN, Xin-Ping; HUANG, Qian. Improving automatic detection of defects in castings by applying wavelet technique. **IEEE Transactions on Industrial Electronics**, v. 53, n. 6, p. 1927–1934, Dec 2006. ISSN 0278-0046.

LIENERT, Thomas; SIEWERT, Thomas; BABU, Sudarsanam; ACOFF, Viola; SPECIFICATIONS, Standard Welding Procedure. **ASM handbook, volume 6A: welding fundamentals and processes**. *[S.l.]*: ASM International Russell, 2011.

LIN, Tsung-Yi; GOYAL, Priya; GIRSHICK, Ross; HE, Kaiming; DOLLAR, Piotr. Focal loss for dense object detection. *In*: **IEEE International Conference on Computer Vision (ICCV)**. *[S.l.: s.n.]*, 2017. p. 2980–2988.

LIU, R.; YANG, B.; HAUPTMANN, A. G. Simultaneous bearing fault recognition and remaining useful life prediction using joint loss convolutional neural network. **IEEE Transactions on Industrial Informatics**, p. 1–1, 2019. ISSN 1551-3203.

LIU, Weipeng; SHAN, Shengqi; CHEN, Haiyong; WANG, Rui; SUN, Jiaming; ZHOU, Zhengkui. X-ray weld defect detection based on AF-RCNN. **Welding in the World**, v. 66, p. 1–13, 03 2022.

LIU, Xiaoyuan; LIU, Jinhai; WANG, Zi; WANG, Lei; ZHANG, Huaguang. Basic-class and cross-class hybrid feature learning for class-imbalanced weld defect recognition. **IEEE Transactions on Industrial Informatics**, IEEE, 2022.

MALAMAS, Elias N; PETRAKIS, Euripides GM; ZERVAKIS, Michalis; PETIT, Laurent; LEGAT, Jean-Didier. A survey on industrial vision systems, applications and tools. **Image and vision computing**, Elsevier, v. 21, n. 2, p. 171–188, 2003.

MANISH, R; VENKATESH, Adigopula; ASHOK, S Denis. Machine vision based image processing techniques for surface finish and defect inspection in a grinding process. **Materials Today: Proceedings**, Elsevier, v. 5, n. 5, p. 12792–12802, 2018.

MARTÍNEZ, S. Satorres; ORTEGA, J. Gómez; GARCÍA, J. Gámez; GARCÍA, A. Sánchez; ESTÉVEZ, E. Estévez. An industrial vision system for surface quality inspection of transparent parts. **The International Journal of Advanced Manufacturing Technology**, v. 68, n. 5, p. 1123–1136, Sep 2013. ISSN 1433-3015.

MERY, Domingo; RIFFO, Vladimir; ZSCHERPEL, Uwe; MONDRAGÓN, German; LILLO, Iván; ZUCCAR, Irene; LOBEL, Hans; CARRASCO, Miguel. Gdxray: The database of x-ray images for nondestructive testing. **Journal of Nondestructive Evaluation**, Springer, v. 34, p. 1–12, 2015.

MIAO, Rui; GAO, Yuntian; GE, Liang; JIANG, Zihang; ZHANG, Jie. Online defect recognition of narrow overlap weld based on two-stage recognition model combining continuous wavelet transform and convolutional neural network. **Computers in Industry**, v. 112, p. 103115, 2019. ISSN 0166-3615.

MIAO, Rui; GAO, Yuntian; GE, Liang; JIANG, Zihang; ZHANG, Jie. Online defect recognition of narrow overlap weld based on two-stage recognition model combining continuous wavelet transform and convolutional neural network. **Computers in Industry**, v. 112, p. 103115, 2019. ISSN 0166-3615. Available at: https://www.sciencedirect.com/science/article/pii/S0166361519301897.

MIAO, Rui; SHAN, Zhangtuo; ZHOU, Qingye; WU, Yizhou; GE, Liang; ZHANG, Jie; HU, Hao. Real-time defect identification of narrow overlap welds and application based on convolutional neural networks. **Journal of Manufacturing Systems**, v. 62, p. 800–810, 2022. ISSN 0278-6125.

Micro-Epsilon. **Automated surface inspection: reflectCONTROL PSS 8005.D**. 2019. https://www.micro-epsilon.com/download/products/dat--reflectCONTROL-PSS-8005-D--en.pdf. Online; accessed 7 July 2019.

NACEREDDINE, Nafaa; GOUMEIDANE, Aicha Baya; ZIOU, Djemel. Unsupervised weld defect classification in radiographic images using multivariate generalized gaussian mixture model with exact computation of mean and shape parameters. **Computers in Industry**, v. 108, p. 132–149, 2019. ISSN 0166-3615. Available at: https://www.sciencedirect.com/science/article/pii/S0166361518305967.

NGUYEN, Hoan-Viet; BAE, Jun-Hee; LEE, Yong-Eun; LEE, Han-Sung; KWON, Ki-Ryong. Comparison of pre-trained yolo models on steel surface defects detector based on transfer learning with gpu-based embedded devices. **Sensors**, v. 22, n. 24, 2022. ISSN 1424-8220.

NOWROTH, Christian; GU, Tiansheng; GRAJCZAK, Jan; NOTHDURFT, Sarah; TWIEFEL, Jens; HERMSDORF, Jörg; KAIERLE, Stefan; WALLASCHEK, Jörg. Deep learning-based weld contour and defect detection from micrographs of laser beam welded semi-finished products. **Applied Sciences**, MDPI, v. 12, n. 9, p. 4645, 2022.

PAN, Haihong; PANG, Zaijun; WANG, Yaowei; WANG, Yijue; CHEN, Lin. A new image recognition and classification method combining transfer learning algorithm and mobilenet model for welding defects. **IEEE Access**, v. 8, p. 119951–119960, 2020.

PATIL, Rajesh; REDDY, Yerreddy. Multiform weld joint flaws detection and classification by sagacious artificial neural network technique. **Int. Journal of Advanced Manufacturing Tech.**, v. 125, p. 1–31, 2023.

POLIKAR, Robi; UDPA, Lalita; UDPA, Satish S; TAYLOR, Tom. Frequency invariant classification of ultrasonic weld inspection signals. **IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control**, IEEE, v. 45, n. 3, p. 614–625, 1998.

REDMON, Joseph; DIVVALA, Santosh; GIRSHICK, Ross; FARHADI, Ali. You only look once: Unified, real-time object detection. *In*: **The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. *[S.l.: s.n.]*, 2016.

REDMON, Joseph; FARHADI, Ali. Yolo9000: Better, faster, stronger. *In*: **IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. *[S.l.: s.n.]*, 2017. p. 7263–7271.

REDMON, Joseph; FARHADI, Ali. YOLOv3: An incremental improvement. **CoRR**, abs/1804.02767, p. 1–6, 2018. Available at: http://arxiv.org/abs/1804.02767.

REN, Zhonghe; FANG, Fengzhou; YAN, Ning; WU, You. State of the art in defect detection based on machine vision. **International Journal of Precision Engineering and Manufacturing-Green Technology**, Springer, v. 9, n. 2, p. 661–691, 2022.

RONNEBERGER, Olaf; FISCHER, Philipp; BROX, Thomas. U-Net: Convolutional networks for biomedical image segmentation. *In*: NAVAB, Nassir; HORNEGGER, Joachim; WELLS, William M.; FRANGI, Alejandro F. (Ed.). **Medical Image Computing and Computer-Assisted Intervention (MICCAI)**. Cham: Springer International Publishing, 2015. p. 234–241. ISBN 978-3-319-24574-4.

SKILTON, Robert; GAO, Yang. Combining object detection with generative adversarial networks for in-component anomaly detection. **Fusion Engineering and Design**, Elsevier, v. 159, p. 111736, 2020.

SONG, L.; LIN, W.; YANG, Y.; ZHU, X.; GUO, Q.; XI, J. Weak micro-scratch detection based on deep convolutional neural network. **IEEE Access**, v. 7, p. 27547–27554, 2019. ISSN 2169-3536.

SUN, J.; WANG, P.; LUO, Y.; LI, W. Surface defects detection based on adaptive multiscale image collection and convolutional neural networks. **IEEE Transactions on Instrumentation and Measurement**, p. 1–11, 2019. ISSN 0018-9456.

SUN, W.; ZHAO, R.; YAN, R.; SHAO, S.; CHEN, X. Convolutional discriminative feature learning for induction motor fault diagnosis. **IEEE Transactions on Industrial Informatics**, v. 13, n. 3, p. 1350–1359, June 2017. ISSN 1551-3203.

SUN, Xiaohong; GU, Jinan; TANG, Shixi; LI, Jing. Research progress of visual inspection technology of steel products—a review. **Applied Sciences**, MDPI, v. 8, n. 11, p. 2195, 2018.

THOMPSON, Donald O; CHIMENTI, Dale E. Review of progress in quantitative nondestructive evaluation. Springer Science & Business Media, 2012.

TRIPICCHIO, Paolo; CAMACHO-GONZALEZ, Gerardo; D'AVELLA, Salvatore; A. Welding defect detection: Coping with artifacts in the production line. **The International Journal of Advanced Manufacturing Technology**, Springer, v. 111, p. 1659–1669, 2020.

WANG, C.; WANG, N.; HO, M.; CHEN, X.; SONG, G. Design of a new vision-based method for the bolts looseness detection in flange connections. **IEEE Transactions on Industrial Electronics**, p. 1–1, 2019. ISSN 0278-0046.

WANG, Chien-Yao; BOCHKOVSKIY, Alexey; LIAO, Hong-Yuan Mark. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *In*: **IEEE CVPR**. *[S.l.: s.n.]*, 2023. p. 7464–7475.

WANG, Xufei; SONG, Jeongyoung. Iciou: Improved loss based on complete intersection over union for bounding box regression. **IEEE Access**, IEEE, v. 9, p. 105686–105695, 2021.

WANG, Xintao; XIE, Liangbin; DONG, Chao; SHAN, Ying. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. *In*: **IEEE/CVF International Conference on Computer Vision (ICCV) Workshops**. *[S.l.: s.n.]*, 2021. p. 1905–1914.

WANG, You; HU, Wentao; WEN, Long; GAO, Liang. A new foreground-perception cycle-consistent adversarial network for surface defect detection with limited high-noise samples. **IEEE Transactions on Industrial Informatics**, PP, p. 1–10, 2023.

WANG, Zhenmin; CHEN, Haoyu; ZHONG, Qiming; LIN, Sanbao; WU, Jianwen; XU, Mengjia; ZHANG, Qin. Recognition of penetration state in gtaw based on vision transformer using weld pool image. **The International Journal of Advanced Manufacturing Technology**, Springer, p. 1–14, 2022.

WEN, Shengping; CHEN, Zhihong; LI, Chaoxian. Vision-based surface inspection system for bearing rollers using convolutional neural networks. **Applied Sciences**, MDPI, v. 8, n. 12, p. 2565, 2018.

WIN, M.; BUSHROA, A. R.; HASSAN, M. A.; HILMAN, N. M.; IDE-EKTESSABI, A. A contrast adjustment thresholding method for surface defect detection based on mesoscopy. **IEEE Transactions on Industrial Informatics**, v. 11, n. 3, p. 642–649, June 2015. ISSN 1551-3203.

WOJKE, Nicolai; BEWLEY, Alex; PAULUS, Dietrich. Simple online and realtime tracking with a deep association metric. *In*: IEEE. **2017 IEEE international conference on image processing (ICIP)**. *[S.l.]*, 2017. p. 3645–3649.

WOO, Sanghyun; PARK, Jongchan; LEE, Joon-Young; KWEON, In So. Cbam: Convolutional block attention module. *In*: **Proceedings of the European Conference on Computer Vision (ECCV)**. *[S.l.: s.n.]*, 2018. p. 3–19.

XU, H.; YAN, Z.H.; JI, B.W.; HUANG, P.F.; CHENG, J.P.; WU, X.D. Defect detection in welding radiographic images based on semantic segmentation methods. **Measurement**, v. 188, p. 110569, 2022. ISSN 0263-2241.

YANG, Lei; SONG, Shouan; FAN, Junfeng; HUO, Benyan; LI, En; LIU, Yanhong. An automatic deep segmentation network for pixel-level welding defect detection. **IEEE Trans. on Inst. and Measurement**, v. 71, p. 1–10, 2022.

YANG, Lei; WANG, Huaixin; HUO, Benyan; LI, Fangyuan; LIU, Yanhong. An automatic welding defect location algorithm based on deep learning. **NDT & E International**, v. 120, p. 102435, 2021. ISSN 0963-8695.

YANG, Lei; XU, Shuai; FAN, Junfeng; LI, En; LIU, Yanhong. A pixel-level deep segmentation network for automatic defect detection. **Expert Systems with Applications**, Elsevier, v. 215, p. 119388, 2023.

YAZDCHI, M.; YAZDI, M.; MAHYARI, A. G. Steel surface defect detection using texture segmentation based on multifractal dimension. *In*: **International Conference on Digital Image Processing**. *[S.l.: s.n.]*, 2009. p. 346–350.

ZENG, XM; LUCAS, J; FANG, MTC. Use of neural networks for parameter prediction and quality inspection in tig welding. **Transactions of the Institute of Measurement and Control**, Sage Publications Sage CA: Thousand Oaks, CA, v. 15, n. 2, p. 87–95, 1993.

ZHANG, Zhifen; YANG, Zhe; REN, Wenjing; WEN, Guangrui. Random forest-based real-time defect detection of al alloy in robotic arc welding using optical spectrum. **J. of Manufacturing Proc.**, v. 42, p. 51–59, 2019. ISSN 1526-6125.

ZHAO, Zhicai; LV, Na; XIAO, Runquan; CHEN, Shanben. A novel penetration state recognition method based on lstm with auditory attention during pulsed GTAW. **IEEE Trans. on Industrial Informatics**, PP, p. 1–11, 2022.

ZHENG, Yangtao; HUANG, Di; LIU, Songtao; WANG, Yunhong. Cross-domain object detection through coarse-to-fine feature adaptation. *In*: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. *[S.l.: s.n.]*, 2020.

ZOU, Q.; ZHANG, Z.; LI, Q.; QI, X.; WANG, Q.; WANG, S. Deepcrack: Learning hierarchical convolutional features for crack detection. **IEEE Transactions on Image Processing (TIP)**, v. 28, n. 3, p. 1498–1512, March 2019. ISSN 1057-7149.