POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

A State of the Art Review of PCB Optical Inspection Methodologies

Tesi di Laurea Magistrale in Computer Science & Engineering

Author: **Alessandro Griffanti**

Student ID: 250114
Advisor: Prof. Loredana Cristaldi
Co-advisors: Luca Martiri
Academic Year: 2025-2026

Abstract

In the fourth industrial revolution context (Industry 4.0), The optimization of production processes and quality assurance have become fundamental. This thesis explores the evolution of automated optical inspection (AOI) methods for printed circuit board (PCB) defect detection, which is of paramount importance in the electronics industry. First, this work analyzes traditional automated optical inspection systems, examining the differences between 2D and 3D acquisition technologies. Although these systems represent the established industry standard, the analysis highlights the inherent limitations of approaches based on rigid rules, which are often subject to high false positive rates and difficulties in adapting to complex scenarios. To overcome these critical issues, the research shifts towards the analysis of techniques based on artificial intelligence. First, Machine Learning (ML) techniques are explored, comparing different algorithms and evaluating their advantages and limitations in terms of accuracy and generalization capabilities. Finally, the paper explores Deep Learning (DL) techniques, highlighting how neural networks have revolutionized PCB inspection. Advanced techniques such as patch classification, semantic segmentation and anomaly detection are analyzed, with particular emphasis on the strategic role of dataset quality. In conclusion, the paper offers a critical overview of the state of the art, highlighting how the transition from deterministic systems to data-driven solutions is key to increasing production efficiency and drastically reducing the margin of error in defect detection.

Keywords: PCB, Automated Optical Inspection (AOI), Industry 4.0, Machine Learning, Deep Learning, Anomaly Detection.

Abstract in lingua italiana

Nel contesto della quarta rivoluzione industriale (Industria 4.0), l'ottimizzazione dei processi produttivi e la garanzia della qualità sono diventati pilastri fondamentali. Questa tesi esplora l'evoluzione dei metodi di ispezione automatica per il rilevamento di difetti nei circuiti stampati (PCB), un componente critico per l'intera industria elettronica. Il lavoro analizza inizialmente i sistemi di ispezione ottica automatizzata tradizionali, esaminando le differenze tra le tecnologie di acquisizione 2D e 3D. Sebbene tali sistemi rappresentino lo standard industriale consolidato, l'analisi evidenzia i limiti intrinseci degli approcci basati su regole rigide, spesso soggetti a elevati tassi di falsi positivi e difficoltà nell'adattarsi a scenari complessi. Per superare queste criticità, la ricerca si sposta verso l'analisi di tecniche basate su intelligenza artificiale. Dapprima vengono esplorate tecniche di Machine Learning (ML), confrontando diversi algoritmi e valutandone vantaggi e limiti in termini di accuratezza e capacità di generalizzazione. Infine, l'elaborato esplora le frontiere del Deep Learning (DL), evidenziando come le reti neurali abbiano rivoluzionato l'ispezione dei PCB. Vengono analizzate tecniche avanzate quali la classificazione di patch, la segmentazione semantica e l'anomaly detection, ponendo particolare accento sul ruolo strategico della qualità dei dataset. In conclusione, l'elaborato offre una panoramica critica sullo stato dell'arte, evidenziando come la transizione da sistemi deterministici a soluzioni data-driven rappresenti la chiave per incrementare l'efficienza produttiva e ridurre drasticamente i margini di errore nella rilevazione dei difetti.

Parole chiave: PCB, Ispezione ottica automatizzata, industria 4.0, Machine Learning, Deep Learning, Anomaly Detection.

Contents

1 | Introduction to image inspection for fault detection

1.1. Industry 4.0

The manufacturing industry and process has been evolving since the 18th century with the first industrial revolution. The second industrial revolution, driven by electrification and mass production, and the third one, characterized by the advent of computers, the internet and automation, carried this sector into the 2010s, when a new phase, the so-called fourth industrial revolution, began [1].

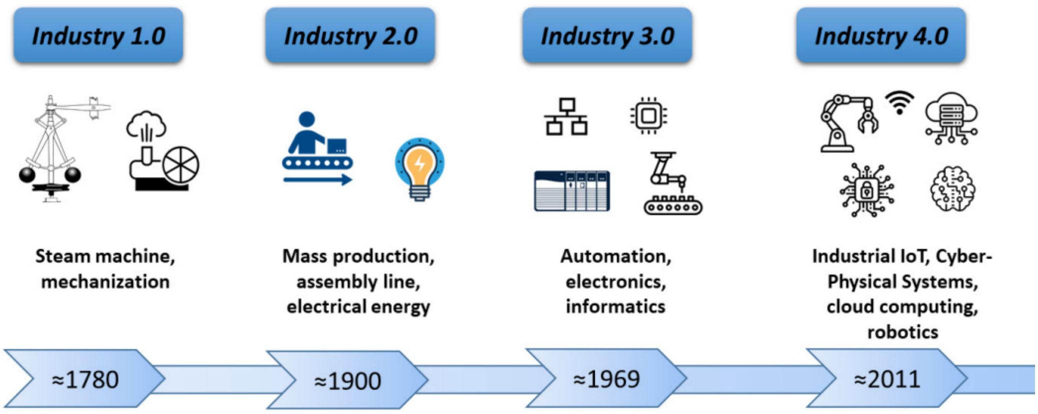


Figure 1.1: Timeline of industrial revolutions [2].

The term 'Industry 4.0' originated in Germany in 2011, with the label 'Industrie 4.0', within a project of the High-Tech Strategy as a policy initiative to accelerate manufacturing computerization and digitalization [3], representing, for the first time, a predicted industrial revolution rather than an a-posteriori observed one [4]. This concept refers to the "intelligent networking of machines and processes for industry with the help of information and communication technology" [5] and is characterized by the advent and use in the industry of technologies such as internet of things (IoT), virtual and augmented reality, cloud computing, big data analytics and, above all, artificial intelligence (AI) with machine learning (ML) and deep larning (DL).

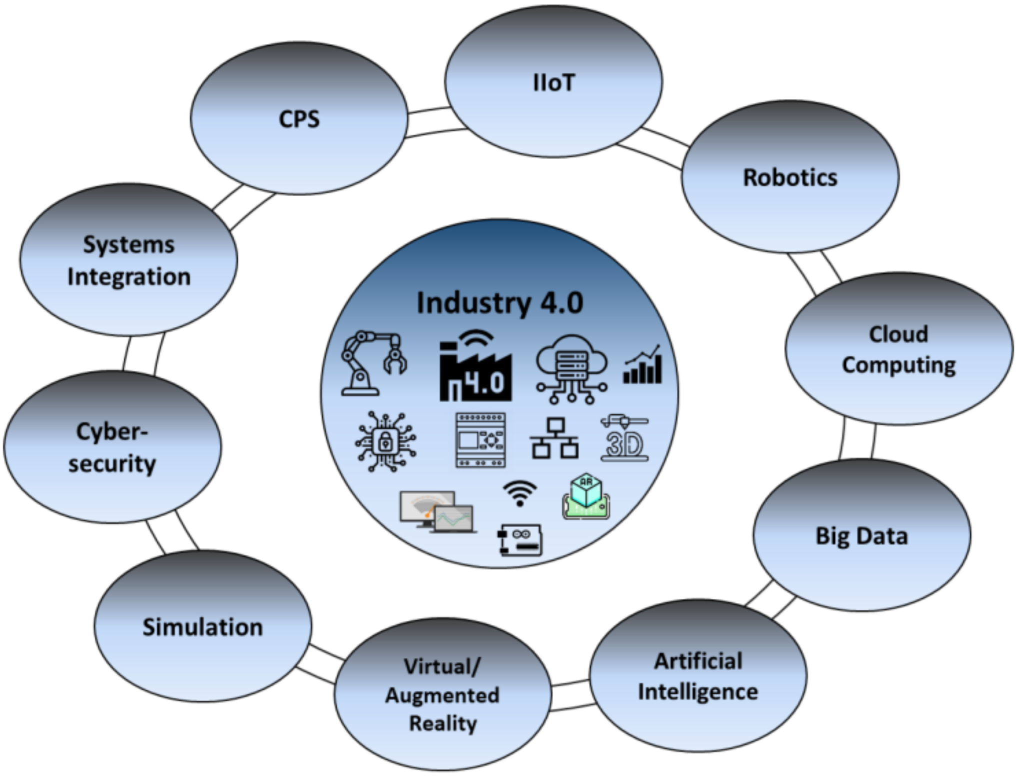


Figure 1.2: Main technologies of Industry 4.0 [2].

Liu et al. [6] analyzed the evolution of quality control, highlighting how it moved from reactive inspections (Quality 1.0), to standards and quality assurance (2.0), Total Quality Management (TQM, Quality 3.0) till Quality 4.0. At this stage, quality control becomes digital, intelligent and predictive thanks to the integration with the already mentioned characterizing technologies of Industry 4.0.

In this setting, among all other innovations, AI represents the key technology enabling advanced automation and autonomous real-time decisions, optimizing production, reducing defects, and improving overall efficiency. Several fields have been influenced and improved by its advent. In quality control, the integration of AI enables automatic in-line inspections with the ability to detect and classify defects in real time, increasing productivity and efficiency compared to traditional systems. As highlighted by Sundaram and Zeid [7], AI is increasingly employed in quality control for tasks such as visual inspection, where deep learning models, such as Convolutional Neural Networks (CNNs), can achieve near-perfect accuracy and outperform traditional inspection methods. In the design field, generative design optimize materials and shapes, reducing costs and development times.

It also contributes to sustainability by reducing energy consumption and production waste and, as said, allowed to transform quality management from reactive to predictive and data-driven [8].

Obviously, the industrial adoption of AI introduces additional challenges. Among the most critical factors are the need for vast amounts of data and, even more importantly, ensuring high data quality. This includes tasks such as accurate data annotation, thorough data cleaning, safeguarding sensitive information and privacy, and ensuring the safety and robustness of AI models [9].

1.2. Machine vision and image inspection for fault detection

Machine vision (MV), a subfield of computer vision, is one of the fields which have been highly impacted by the integration of AI in the industry [10]. Its primary goal is to replicate the human sense of sight in machines through digital cameras, sensors and algorithms and, by doing so, to enable machines to perceive and interpret their surroundings, through the acquisition and automatic processing of images [11].

In Industry 4.0, machine vision plays a central role as an enabling technology, as it allows for the automatic acquisition and interpretation of images to improve the efficiency, precision, and autonomy of industrial processes. Among the main industrial uses, summarized in Figure 1.3 [12], the most relevant is quality control (29%), followed by process monitoring (24%) and detection purposes (21%). These applications, which mainly concern manufacturing and automotive industries, not only allow for the identification of defects and anomalies in real time, but also enable the guidance of robots, inspection of surfaces, reading of codes, and management of automatic product sorting [12].

As for fault and defect detection, Ren et al. [13] classify in three classes the main tasks based on MV: classification, localization and segmentation. The first one refers to the presence or absence of a defect in an image. The second one refers to the exact localization of the defect in a given image, while segmentation can be used to retrieve important information concerning the defect severity.

In this context, Automated Optical Inspection (AOI) represents the most mature application of machine vision for quality control and defect detection in Industry 4.0. It employs an hardware-software chain, described in detailed in chapter 2, which enables non-destructive acquisition and automatic processing of images [8]. As shown in Figure 1.4 [12], AOI is deployed across many sectors: in automotive, for the automatic

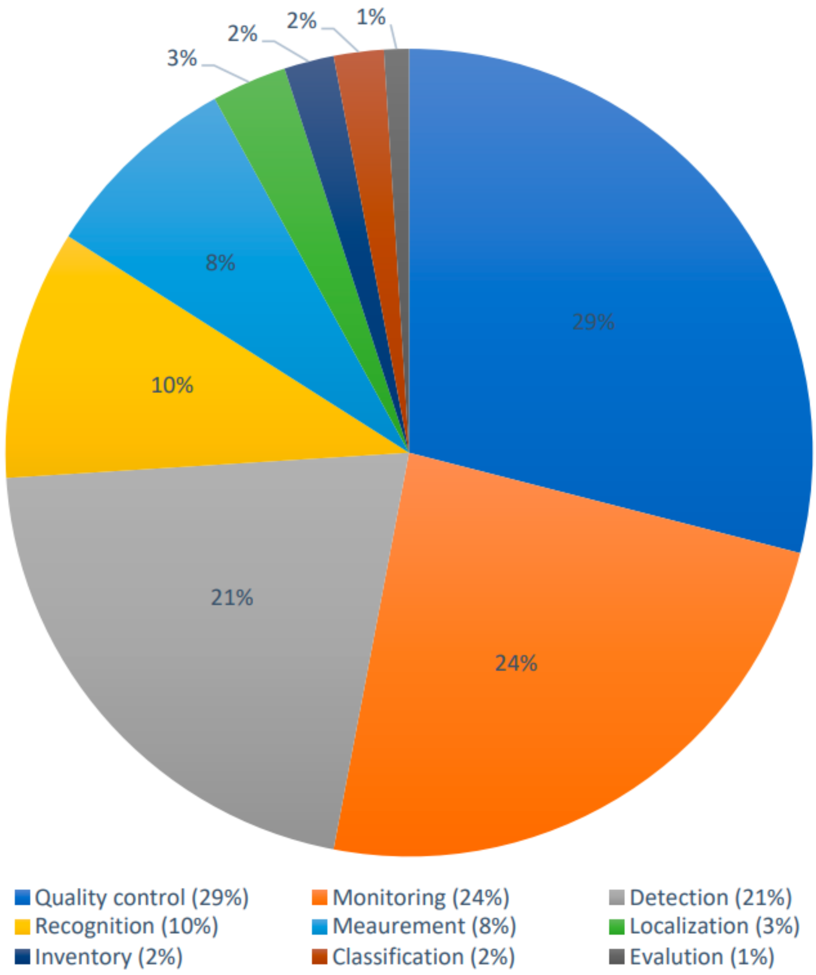


Figure 1.3: Use of Machine Vision in industry 4.0 [12].

inspection of surface defects on car bodies (e.g., scratches and dents) [14]; in textiles, where it replaces manual checks and detects defective weaves, pulled threads, and holes directly at the loom, improving quality and throughput [15]; in the pharmaceutical industry, to automate the quality control of tablets and blister packs and to identify damage, color deviations, and other defects [16]; In the energy sector, to detect coating defects of lithium-ion batteries electrodes [17] and to identify multiple defect types in photovoltaic modules [18]; in display manufacturing (TFT-LCD), to rapidly find panel defects on the line [19]; moreover, in glass containers, AOI checks chips in the mouth of glass bottles and dimensional defects [20].

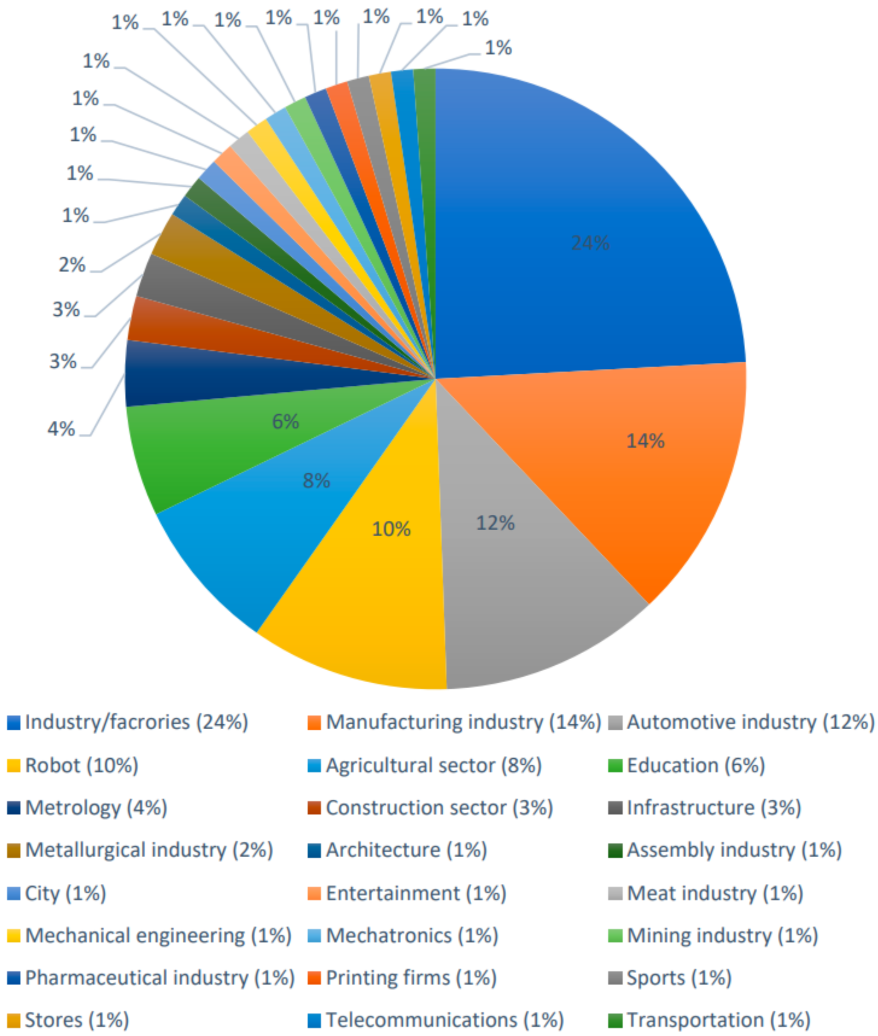


Figure 1.4: Application sectors of machine vision in industry 4.0 [12].

Among its great advantages, AOI addresses the well-known limits of manual visual checks. Human inspectors are slow, their performance is variable, and accuracy declines on repetitive tasks. As it was already clear at the beginning of its development in the 1980s for

Chin and Harlow [21], AOI offers the alternative: it eliminates human labor from routine visual checks, provides speed and diagnostic capabilities, can operate in unfavorable environments, reduces the demand for highly skilled inspectors, enables statistical analysis and record-keeping to support management decisions, and matches high-speed production with high-speed inspection.

1.3. PCB defect detection

Among the fields in which AOI finds applications, one of the most important is represented by the manufacturing and electronics industry and, in particular, by the defect detection in Printed Circuit Board (PCB).

A PCB, Figure 1.5 [22], is a thin board that contains etched or printed conductive pathways. These pathways electrically connect various electronic components, such as transistors, resistors, and integrated circuits, which are soldered onto the board. The typical structure (stack-up) consists of layers of copper separated by an insulating substrate and covered with a protective solder mask, with silkscreen printing for text and reference symbols. There are single-sided, double-sided, and multilayer boards. The layers are connected by vias (through-hole, blind, buried) and components are mounted using through-hole technology (the pins pass through the board) or Surface Mount Technology (SMT) [22].

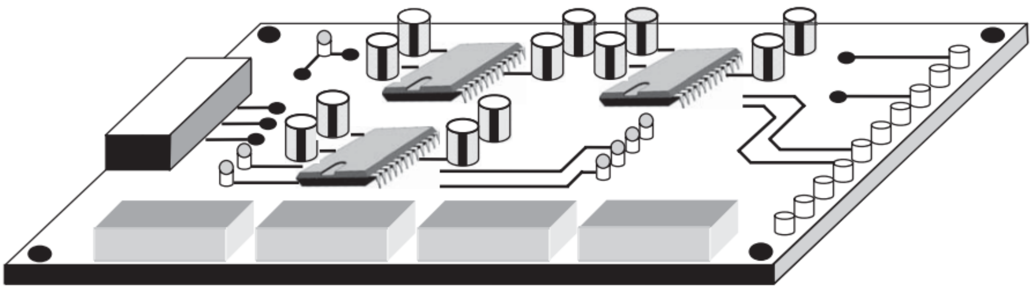


Figure 1.5: PCB graphical representation [22].

PCB are the essential components in several electronic devices and, as a consequence, detecting their defects is a capital task. Any non-detected faults can compromise their functioning and cause the failure of entire electronic systems. Early detection of such defects not only prevents catastrophic consequences, but also contributes to reducing costs and increases product reliability [23].

The main PCB inspection techniques employed nowadays cover both superficial and hidden defects. AOI is the key tool for detecting the first category of defects, while X-ray, thermography and ultrasonic testing are employed to detect internal defects and other problems not easily detectable with AOI. On the electrical front, ICT, flying-probe and boundary-scan/JTAG are treated as high-coverage, low-time solutions for miniaturized boards [23, 24].

This thesis focuses on AOI for PCB. First, chapter 2 deals both with traditional 2D AOI systems, describing the hardware components, the key techniques and algorithms used and the main detectable defects with such technologies, and with the more advanced 3D AOI systems, highlighting the differences with traditional techniques and the limits that made such an improvement necessary. Then, chapter 3 describes how machine learning and deep learning have improved this field and the approaches based on those technologies.

2 | Automated optical inspection applied to PCB

In this section, it is presented the main technology employed to detect defects in PCB: AOI. The different available systems and the phases of the process, with a particular focus on how the images are acquired, are being carefully described. Lastly, they are presented detection techniques and algorithms based on image processing and the main defects detectable with these techniques.

2.1. AOI for PCB: an overview

The *inspection* process has existed since the beginning of industrial production and it refers to the procedure of identifying defects of a product with respect to a certain model or prototype [25]. This essential practice has evolved significantly over time and its automation is what we nowadays call *AOI*, a non-distructive quality assurance method relying on visual imaging techniques that is widely used in the electronics industry, especially for PCBs inspection [26].

With the advent of new techniques and algorithms, machine learning and finally deep learning, AOI systems have been significantly reshaped over time, but the core ideas and principles, originating in the 1980s, are still valid. Among these are the main phases of this process: image acquisition, image processing, and output [27]. While these stages have been preserved, recent developments have refined them into a more structured workflow, as described in [28], consisting of:

- **Image acquisition:** In this phase, high resolution images of the PCB under analysis are captured. This topic will be discussed in much more detail in subsection 2.2.1.
- **Image preprocessing:** In this stage, images are enhanced, calibrated and transformed, for example reducing noise and distortions, to allow faster processing.
- **Image analysis and defect detection:** The AOI system processes the images

to discover potential defects in the PCB under inspection. Several techniques and algorithms have been developed over time and will be examined in subsection 2.2.2.

- **Decision making and classification:** Potential PCB defects are classified according to IPC-A-610 standard, the most widely adopted regulation in the electronics industry, into three categories: *acceptable condition*, *process indicator condition* and *defect condition*. Despite their extensive deployment, traditional AOI systems still present notable limitations, discussed in section 3.1, leading to the continued need for manual inspection and verification.

AOI systems can be classified into three main classes [28]:

- **2D AOI systems:** They represent the most adopted and widely used category of AOI systems. They rely on the acquisition of high-resolution bidimensional images, typically using a single camera and a controlled illumination environment.
- **3D AOI systems:** One of the primary problems of traditional 2D AOI systems is their inability to acquire information related to the z-axis, thus making them ineffective in detecting complex defects requiring volumetric analysis. This limitation is overcome by 3D AOI systems employing the latest technology improvements including data acquisition techniques using multiple cameras, laser triangulation or structured light projection to capture height information. More about these two classes of AOI systems is discussed in the following section 2.2 and section 2.3.
- **AI-driven AOI systems:** This class employs machine learning and deep learning techniques to enhance performance and overcome the limitations of traditional optical inspection. A dedicated chapter 3 explores this category in detail.

2.2. Traditional 2D AOI systems

This section provides an analysis of the traditional 2D AOI systems with a focus on the most important stages: image acquisition and traditional processing techniques and algorithms. Lastly, it is presented a taxonomy of the main defects detectable with such approaches.

2.2.1. Images and acquisition

The image acquisition process is the first and probably most important step in the defect detection pipeline previously described, since the quality of the acquired images directly and inevitably influences the subsequent stages.

Traditional 2D AOI systems are based on a hardware configuration designed to capture high-quality images for defect detection. At the core of this configuration are three essential elements: a camera for image acquisition, a lens and a dedicated illumination source and scheme to provide controlled lighting conditions [29].

Camera

The industrial camera is the key hardware component of the configuration.

In traditional systems, it is typically a single unit operating in a top-view configuration, placed at a 90° angle with respect to the PCB under inspection, employing *Charge-Coupled Device* (CCD) or *Complementary Metal–Oxide–Semiconductor* (CMOS) as image sensor [30].

In traditional PCB AOI systems, camera resolutions reported in the literature range from XGA area-scan sensors (1024×768) to multi-megapixel area sensors (5 MP), and also include line-scan sensors with several thousand pixels per line [31, 32]. However, that does not correspond to the effective inspection precision.

As a matter of fact, beside the sensor resolution, it is also important to consider the system resolution, which represents the minimum size of a defect or feature that the entire acquisition system is able to distinguish. Among the other main factors that contribute to this parameter, there is the working distance (WD), representing the distance between the camera lens and the PCB, and the field of view (FOV), indicating the area of the PCB that can be captured with a single image, which is in turn influenced by the sensor size (SZ). A wide FOV increases coverage per shot but reduces the level of detail, whereas a narrower FOV improves precision at the cost of requiring more acquisitions [33, 34]

Figure 2.1 [34] clearly visualizes the setup and parameters just described.

Lens

While the camera is the receptor of the image, in a machine vision system the lens acts as the eye that produces such an image. The role of this component is similar to the one the crystalline lens plays in the human eye: focus incoming light to produce a clear image. An inappropriate lens choice can result in distortions, loss of detail, or insufficient coverage, severely affecting the accuracy of defect detection [35].

The main parameter that defines the lens is the focal length (FL), which is defined as the distance between the optical centre of a lens and the point where light rays converge to

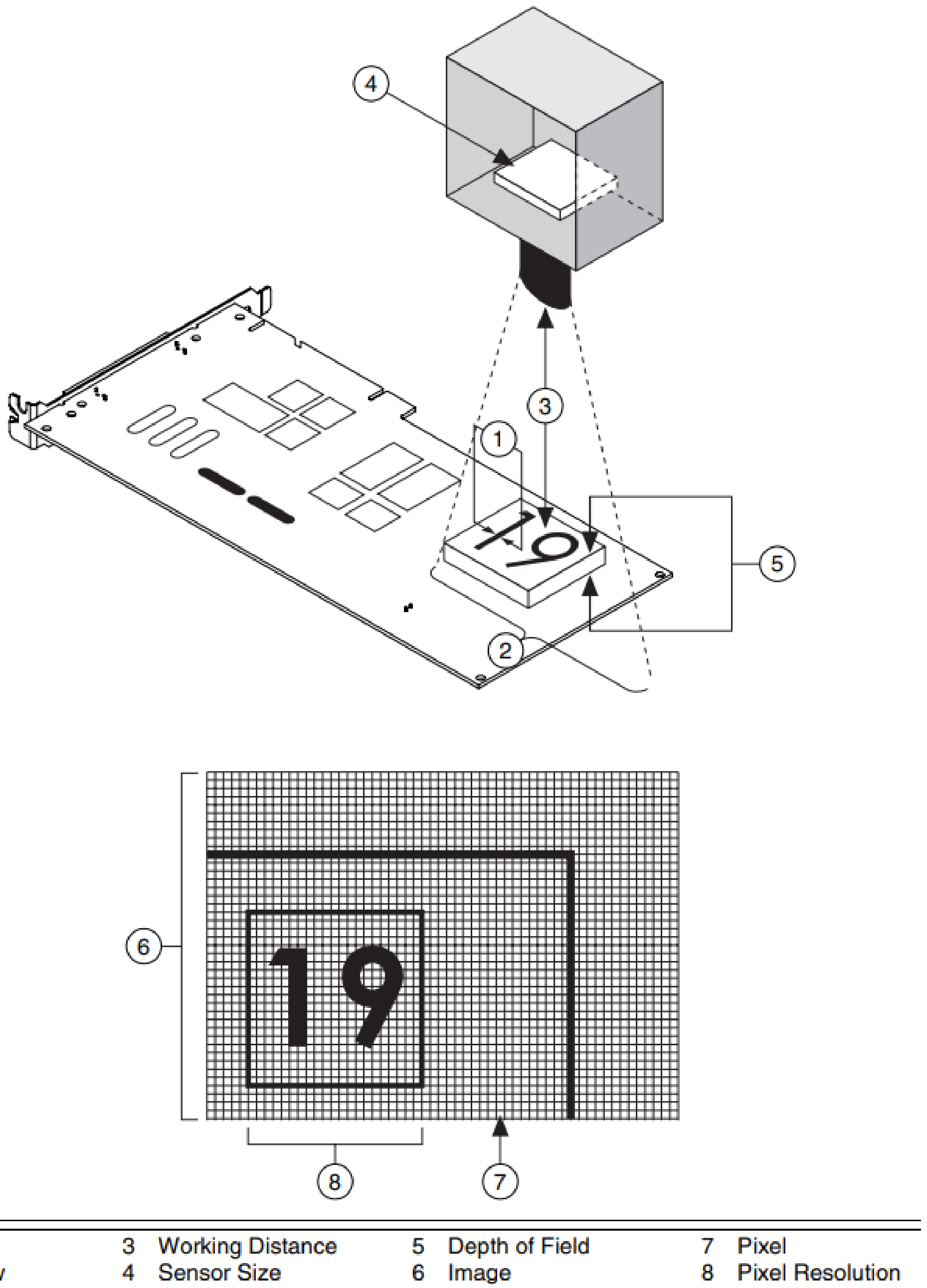


Figure 2.1: Fundamental Parameters of an Imaging System [34].

form a sharp image. The FL [34] is defined as:

$$FL = \frac{SZ \times WD}{FOV}$$

Which highlights how, for a fixed sensor size, increasing the focal length reduces the FOV, while decreasing the focal length expands the coverage at the cost of reduced detail.

Figure 2.2 [29] clearly shows what described.

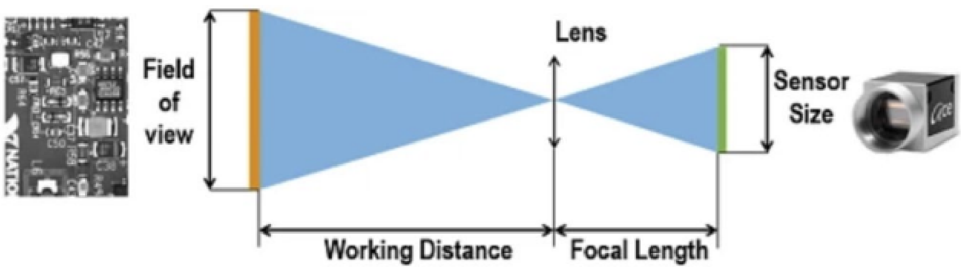


Figure 2.2: Relation between focal length, field of view and working distance [29].

Illumination sources and scheme

The illumination sources and scheme, which refers to how they are organized with respect to the object, play a crucial role in determining the quality of the captured image and, hence, of the inspection process. The reason is that, although sensor and lens parameters are fundamental, all camera sees is the light the PCB under inspection reflects. Therefore, enhancing the contrast of the relevant features is essential. The main factors that influence the choices concerning the illumination setup are the size and material of the object under inspection, its geometry and the inspection environment [29, 36].

As for the illumination sources, nowadays the most widely adopted is *light-emitting diodes* (LED) due to its versatility, lifespan and ease of integration. Beside that, various other sources can and have been used in the past, such as fluorescent, fibre optic halogen, incandescent and xenon lamps [33].

Illumination sources can have different physical shapes, each associated to different applications and illumination scheme. The main ones [36, 37] are:

- **Ring light:** Positioned around the lens, they provide a uniform illumination and are usually adopted in the bright-field configuration.

- **Strip light:** They are suited to illuminate wide areas and can be adopted both in bright-field and dark-field configuration.
- **Dome light:** They generate omnidirectional diffused light, which is ideal to reduce specular reflections from shiny surfaces. They are associated with diffuse lighting schemes.

The second crucial factor is the illumination scheme, which defines how the illumination source, the PCB and the camera are going to interact with each other. Zhou et al. [37] report that the most adopted schemes are:

- **Bright-field illumination:** The light is directed towards the PCB with an angle that allows most of the reflected light to enter the camera.
- **Dark-field illumination:** The light is directed towards the PCB with a low angle and only some reflected light enters the camera.
- **Backlight illumination:** The source is placed behind the PCB.
- **Coaxial illumination:** The light is projected along the optical axis of the camera through a semi-reflective mirror.
- **Diffuse reflection illumination:** Light gets diffused by a spherical surface before hitting the PCB.
- **Structured light illumination:** It projects patterns on the object, allowing to retrive 3D information. This is less common for traditional 2D AOI systems.
- **Stroboscopic illumination:** It involves emitting high-frequency light pulses onto an object.

Figure 2.3 [37] clearly visualizes the different schemes.

2.2.2. Detection techniques and types of detectable defects

In classical literature, the detection techniques used in traditional 2D AOI systems are divided into three main approaches: *referential*, *non-referential* and *hybrid methods* [38, 39]. Referential methods, also known as comparison-based, are based on the comparison between the image of the PCB under inspection and a defect-free PCB image called golden template. Non referential methods, also known as design-rule inspection methods, work without using a reference image and directly verify the compliance of the PCB under inspection to design specification standards. Hybrid methods try to combine these two approaches.

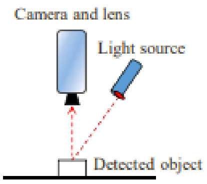
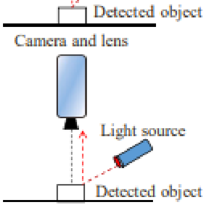
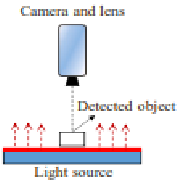
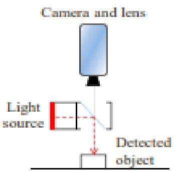
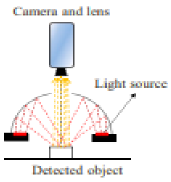
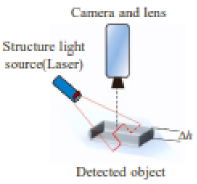
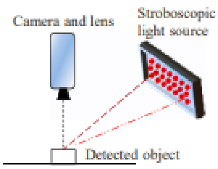
Illumination schemes	Optical pathway diagram	Description	Advantage	Limitation
Bright-field illumination		• The light source has a certain angle with the object to be detected, and most of the reflected light enters the camera directly [40].	✓ High image contrast; ✓ Easy installation	• Uneven distribution of light; • Only detect low-reflective targets.
Dark-field illumination		• Provide low-angle illumination and some specific reflected light into the camera [41].	✓ Excellent feature enhancement performance.	• Low imaging resolution and poor inspection accuracy.
Backlight illumination		• The light source is located on the back of the target to be detected, and the incident light is mainly used for image analysis [42].	✓ High-intensity backlighting; ✓ Strong image contrast.	• Easy to lose surface details.
Coaxial illumination		• After passing through the lens, the light rays illuminate the target along the coaxial direction of the camera [41].	✓ Eliminate the shadow caused by the uneven surface of the object.	• Easy to produce a duplicate image.
Diffuse reflection illumination		• The light irradiates the target through the continuous diffuse reflection of the spherical surface, and the light diffuses into the camera [39].	✓ Large angle of the light source; ✓ Uniform illumination.	• Low light intensity and imaging contrast.
Structured light illumination		• The grating or line light source is projected onto the measured object, and the three-dimensional information of the measured object is obtained according to the imaging distortion [39].	✓ Suitable for detecting the height and depth information of the detection target.	• Expensive
Stroboscopic illumination		• Irradiate a high-frequency light pulse onto an object [43].	✓ High-speed imaging and increased depth of field.	• Energy dissipation.

Figure 2.3: Different illumination schemes and their applications [37].

Referential methods

As said before, referential methods, also called comparison-based, are based on the pixel by pixel comparison against a golden sample, also called template image.

He et al. [40] describe the easiest implementation of this method, image subtraction, and the two main ways in which it can be implemented:

- **XOR operator:** In this approach the images are converted to grayscale and then binarized, which allows, after a precise alignment between them, the application of the XOR operator to obtain a difference image which allows to find the defects.
- **Direct subtraction:** This method works by directly subtracting the pixel values of the two images, eliminating the need for any binarization step. Typically, grayscale images are employed but RGB ones can be used as well to inspect the color distributions in the difference image [41].

Despite being an effective method mainly due to its low computation cost, which is one of the reasons it represents one of the main used methods, image subtraction has several drawbacks. First, it works well when the inspected image is as identical as possible to the golden template. Moreover, it can be used to inspect only PCB of the same type of the template. Finally, it requires precise alignment, size matching and it is sensible to noise and variation of the light conditions, which often require a preprocessing step like Gaussian blurring which can increase the overall complexity.

The other main referential method described by He et al. [40] is template matching, which evaluates the similarity between portions of the image and a golden template using metrics like mean square error (MSE), Structural Similarity Index (SSIM) or Normalized Cross-Correlation (NCC). The idea of this approach, which uses grayscale images to reduce the computation overhead, is to let the template shift across the image to inspect and compute, at each iteration, the similarity between the template and the corresponding image region. When the template is expected to meet a specific feature of the PCB, the similarity result should return a high value. When this value is below a threshold, it means that a defect is present. Also this approach presents limitations, mainly being its sensitivity to rotations, scale and illumination variations.

Beside the image comparison techniques just described, the other main branch of referential methods is represented by model-based inspection techniques, finding in the graph-matching methods their main representative [38]. The core idea of these methods, which is based on the geometrical and structural properties of the binary image and is implemented with attributed graphs [42] and pattern attributed hypergraph [43], is to represent PCB layouts as sets of nodes and spatial relationships. The defects are exposed by comparing the structure of the graph obtained by the PCB under inspection with the one taken as reference [39].

Non-Referential methods

Non-referential methods base their detection on verification of PCB design rules without direct comparison with a golden template. Different techniques have been proposed in the literature. Morphological processing is one of the main non-referential approaches and it is based on the iteration of the expansion-contraction process [39]. Ye e Danielsson [44] proposed an algorithm belonging to this class called *skeleton detection algorithm*. The core idea of this method is the following: if a trace has a width lower than a certain threshold d, then it will disappear if a sequence of shrinking operation is applied to it. More in detail:

1. A shrinking operation is applied to the binary PCB image (a) for $d/2$ iterations (b).
2. It is performed the *Connectivity Preserving Shrinking* (CPS) operation which, unlike classic shrinking, does not eliminate those pixels that are essential for preserving the continuity of a shape (c).
3. The two resulting images are compared using the AND logical operator. The emerging regions are the defect ones, because they are not respecting the minimum width required.

Another approach belonging to the non-referential methods, extensively described by Moganti et al. [39], regards encoding techniques, which employ binary images to operate. Among the two main encoding methods there are boundary analysis and run-length encoding.

The former is based on the simplified representation of traces borders in PCB, using directional encodings such as the *Freeman Chain Code*, which describes borders using sequences of oriented vectors. Once the boarder is encoded, the Euclidean distance and the distance along the contour between pairs of points are compared: anomalous differences indicate potential defects.

The latter analyses the continuous sequences of conductive pixels along the rows and columns of the binary image. Histograms, which represent the frequency of the different lengths of these sequences, called runs, are built for each direction. Very short runs may indicate defects such as interruptions or abnormal narrowing of the tracks [45]. The method thus allows verification of whether the width of the conductors meets the minimum

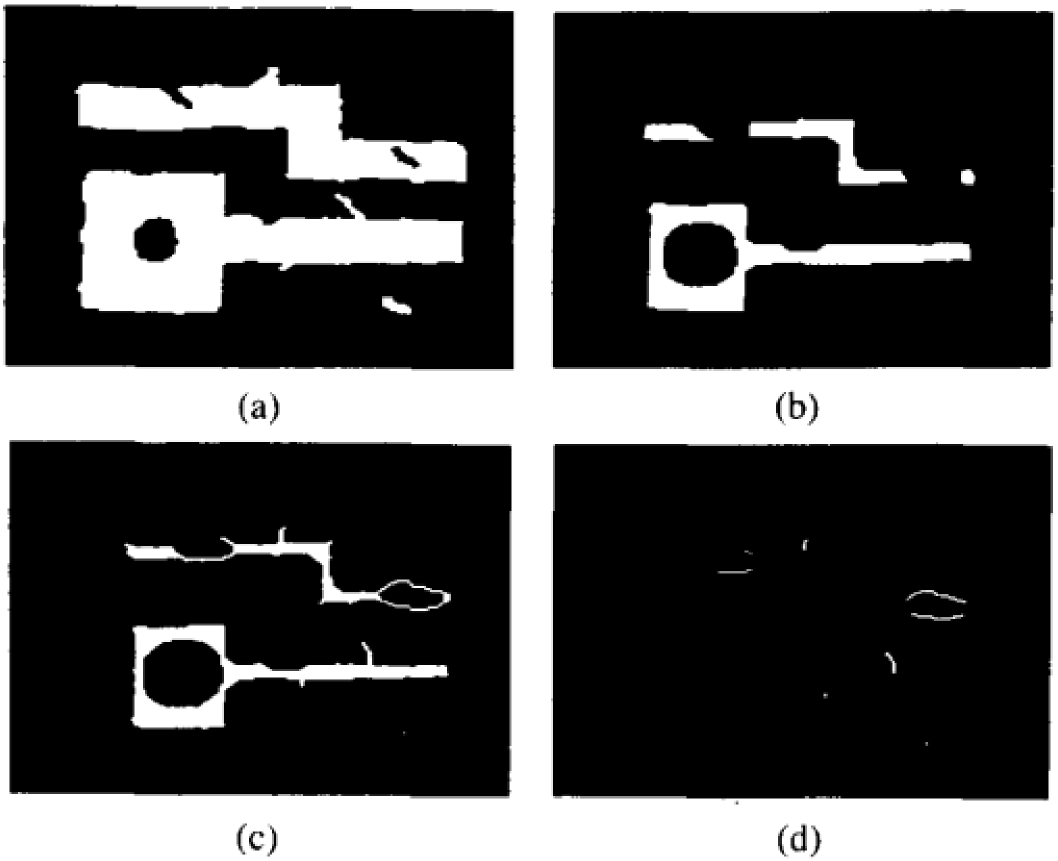


Figure 2.4: Phases of the Skeleton Detection Algorithm [44].

requirements by comparing the lengths of the runs with predefined thresholds.

Despite being effective methods for the detection of certain types of defects, mainly dimensional violations (like thin traces), the main drawback of non-referential methods is that they can't identify defects which are not violating PCB design rules [38, 39].

Hybrid methods

Hybrid methods are based on the idea of combining the referential approach, best suited for certain classes of defects, such as missing components, with the non-referential one, which mainly focuses on other categories of defects, such as traces width. By that, the limits of classical referential and non-referential methods are overcome and a wider spectrum of defects is identified [38, 39].

In this framework, Mandeville [46] proposed the *generic method* which combines the two approaches by comparing not the whole images but a list of detected features on the PCB board with a list of expected feature from the project [39]. This idea allows to overcome a lot of false positive detected by the single approaches.

A second hybrid approach combines boundary analysis and ad-hoc measurements. First, edges that might have defects are selected, and only on these are measurements of thickness and spacing performed [39].

Indera Putera et al. [47] combined morphological processing (non-referential approach) with image subtraction (referential approach) to detect 13 defects and also classify them in 7 categories.

Detectable Defects

In AOI for PCB, defects are mainly divided into two categories: bare PCB (BPCB) and assembled PCB (PCBA). The BPCB is the board made of insulation material with no components, comprising only conductive tracks, microholes/pinholes, metallised holes and solder pads. At this stage, topological, geometrical and process defects prevail. PCBA, on the other hand, concerns the assembly stage, when the components are assembled and soldered onto the board. At this stage, positioning and joining/soldering faults dominate [48].

In Table 2.1, a taxonomy of the different defects and the techniques best suited to detect them is presented.

2D AOI Method	Bare PCB / PCBA	Main defects detectable	Main references
Referential methods	Mainly Bare PCB.	Missing tracks; missing termination; opens and shorts; cuts; minimum width violations; pin holes; missing holes; wrong hole size; misaligned hole; under/over etch; mouse bites; spurs. In addition, missing components and misaligned parts for PCBA.	[38–40]
Non-referential methods	Bare PCB and PCBA.	Opens/partial opens; minimum trace spacing; scratches and surface nonconformities; minimum land width (MLW); minimum conductor spacing (MCS); minimum conductor trace width (MCTW); nicks and bumps; angular errors; annular ring violations. For PCBA: spurious copper; solder paste scooping; solder joint defects; solder volume insufficiency/excess; tombstoning.	[38, 40, 49, 50]
Hybrid methods	Both Bare PCB and PCBA.	Combine both strategies: missing and extraneous features; pads; trace-to-pad connections; isolated blobs; open circuits; excessive trace width; cut solder joints; narrow traces; poor spacing.	[38, 39, 50]

Table 2.1: Main defects detectable with traditional 2D AOI techniques.

2.3. Advanced 3D AOI systems

As said in the previous sections, one of the primary problems of traditional 2D AOI systems is their inability to acquire information related to the z-axis. In particular, certain post-soldering anomalies, defects such as pins or components that are raised (lifted leads/chips), or also particular solder joints defects, are difficult to detect with the simple top view configuration employed in traditional 2D techniques [33].

To overcome these problems, 3D AOI is being implemented in the industry, which empowers three-dimensional measurement to optical inspection.

Li et al. [51] classify 3D optical vision techniques in *passive methods* and *active methods*. Passive methods do not project any addictive signal on the object but retrieve three-dimensional information by analyzing ordinary images. A typical example is stereovision, in which two or more cameras capture the PCB from different angles and, using image matching algorithms, the 3D coordinates of the corresponding points are triangulated [52].

On the other hand, active methods emit structured signals to facilitate 3D reconstruction. This approach is way more employed in the industry because of the higher degree of precision it can reach compared to the passive methods.

Among the active methods, one of the most important is the *structured light method*, in which, as shown in Figure 2.5 [53], a projector sends known light patterns onto the object, which, when deformed on the PCB, allow its topography to be calculated [54].

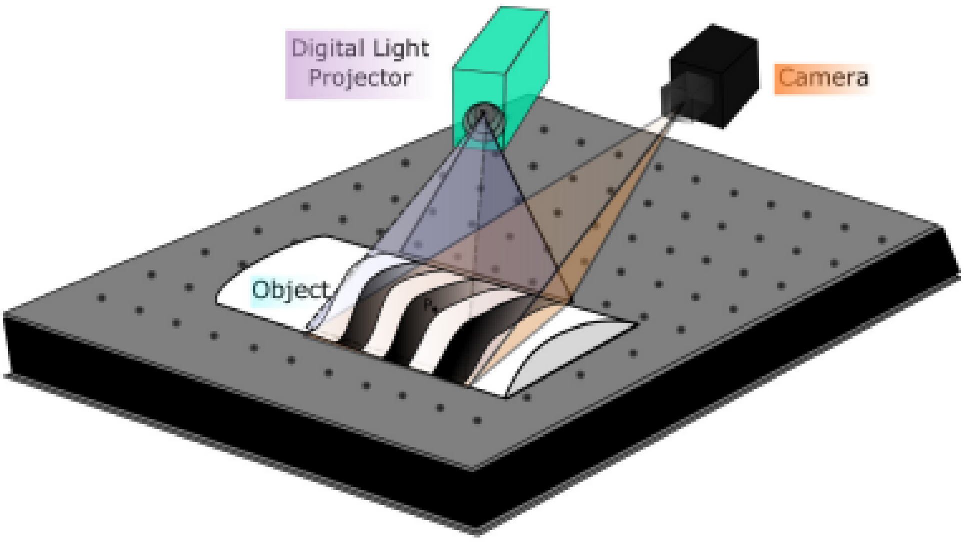


Figure 2.5: Structured light scheme representation [53].

Despite all the different patterns that have been discussed in the literature, *fringe patterns*, which employ sinusoidal patterns, turned out to be among the most effective. Fringe patterns used with the structured light method is called *digital fringe projection* (DFP) [51]. In this system, better depicted in Figure 2.6 [55], vertical lines (fringes) are created (left image) and projected on the object. Those fringes get deformed since they slide over bumps and depressions in the surface (upper image) and the camera captures this deformed version (right image). A software, knowing the relative position between camera and projector and where the fringes should have been, can elaborate the 3D map of the object under inspection [51, 55].

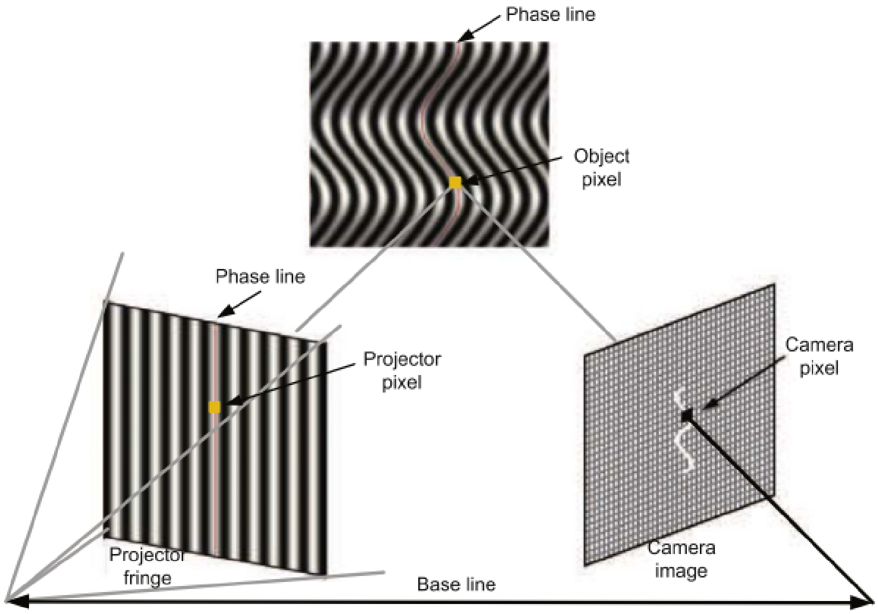


Figure 2.6: Digital fringe projection system [55].

Another relevant active method is laser triangulation. In this technique, a thin laser beam or line, generally placed perpendicular to the object under inspection as shown in Figure 2.7 [56], is scanned across the PCB. The camera observes the deflected position of the line and calculates the height point by point (optical triangulation principle) [51].

Lastly, one class of active methods employs controlled variations of illumination. For example, Vitoriano and Amaral [57] describe a photometric system that reconstructs the 3D shape of the solder joints from 2D images acquired with multi-angle/colour LED ring, which proves to be effective at assessing if the solder joint reaches at least 25% of the component electrode height as requested from IPC-A-610, which 2D AOI systems can't verify.

3D AOI techniques have proved to be highly effective in detecting solder joint defects, as well as inclined or lifted components and other issues that 2D AOI systems cannot reliably identify. This technology is continuously evolving and, alongside X-ray inspection and thermography used to detect internal defects [30], it represents one of the most promising approaches in the field.

As discussed throughout the previous chapters, traditional AOI methods offer several key advantages in industrial PCB quality control. They operate with high efficiency, deliv-

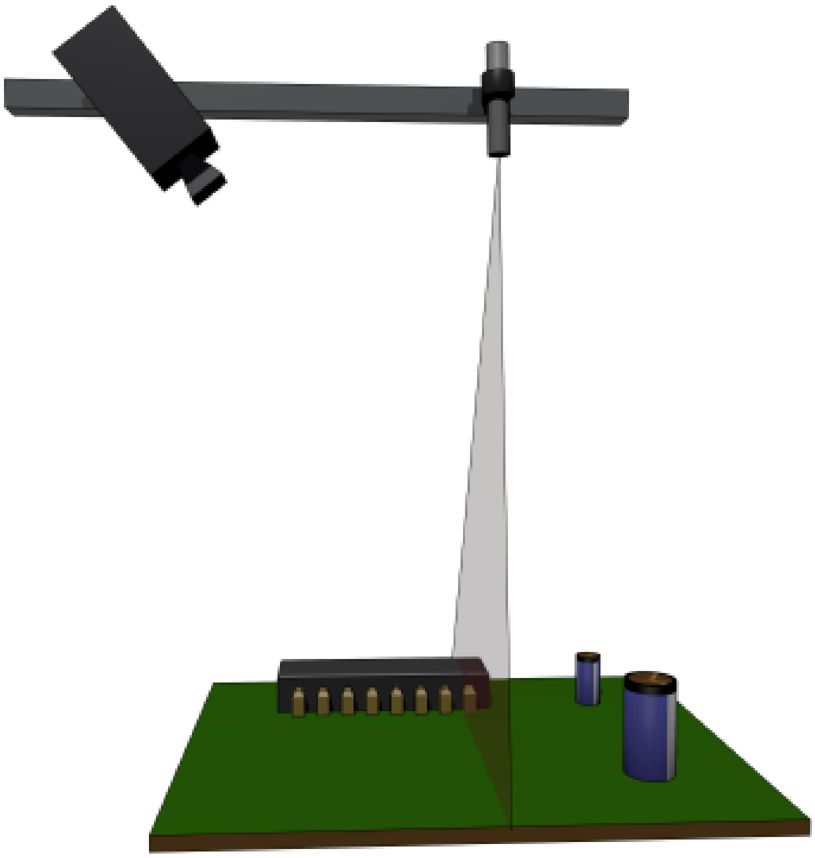


Figure 2.7: Classical laser triangulation setup [56].

ering rapid in-line inspection throughput while maintaining consistent, repeatable results free from human variability. The architecture of classical AOI systems is inherently modular and scalable, allowing configurations to be adapted to different board types and production speeds, a design that supports real-time inspection even in fast-paced, high-volume manufacturing environments. Moreover, many fundamental AOI algorithms (e.g., simple image subtraction) are computationally lightweight, incurring very low processing overhead and thus enabling defect detection with minimal hardware requirements. After decades of development, AOI technology has matured into a reliable, widely adopted solution, with its core principles (originating in the 1980s) still valid and seamlessly integrated into modern high-speed production lines. Each of these characteristics, efficiency, consistency, modularity, low computational cost, and proven maturity, prove the suitability of traditional AOI for real-time industrial inspection [58].

3 | Machine Learning techniques for PCB defect detection

In the following chapters, different techniques with respect to traditional AOI systems are being investigated. In particular, the focus will be on machine learning and deep learning techniques and on both the advantages and disadvantages they bring with respect to the techniques already explored.

3.1. Limitations of traditional AOI methods

Traditional 2D AOI systems, despite being extensively employed in the industry for defect detection and quality inspection of PCB, present several limits and drawbacks which have been widely discussed and faced in the industry.

The first and probably main limitation of this technology is the high false positive rate. A false positive is a negative sample which is incorrectly classified as positive [59], which means, in the quality inspection context under analysis, that a defect-free or compliant PCB is classified as defective. As described in subsection 2.2.2, the most used detection techniques belong to the referential methods. Among the downsides of this approach, there's the lack of flexibility: any unexpected variation, even if within acceptable tolerances, can lead to incorrect alerts. This rigidity, especially with the use of strict thresholds to prevent errors, make inspection vulnerable to variations in lighting, shadows and slight variations in placements, causing many compliant PCB to be labeled as defective [60, 61]. Comparison based techniques effectiveness critically depends on fine alignment indeed: in its absence, subtraction and XOR operations highlight spurious differences that easily result in false alarms [62], as shown in Figure 3.1.

Given the limitations just introduced, it's clear that another important issue is that traditional AOI systems can't guarantee a fully autonomous inspection. It is essential a manual revision of the defects found by the AOI systems in order to distinguish the real defects and the false positives. Specialized operators must therefore visually verify the

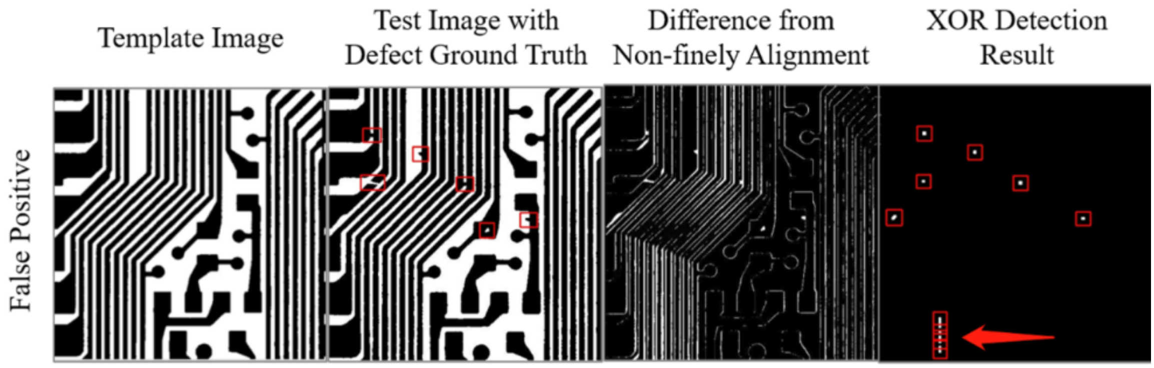


Figure 3.1: False positive example [63].

reported anomalies increasing the human workload despite the adoption of an automatic system. This additional step not only slows down the production flow and partially negates the benefits of automation, as the reliability of the final result still depends on the judgment of a human inspector [60, 64], but it also introduces additional subjectivity in the standards for judgment during the examination [65].

Lastly, as said in section 2.3, traditional 2D AOI systems present limitations also in the spectrum of defects that they can recognize. To overcome this, in addition to the 3D advancements already described, also machine learning and deep learning techniques have been employed and integrated in this field and will be analyzed in the following chapters.

3.2. Machine learning based approaches

ML is a subfield of AI that focuses on enabling systems to learn from data and improve their performance on specific tasks without being explicitly programmed [66]. Although the terms AI, ML and DL are often used interchangeably, they represent different levels of abstraction. In particular, ML and DL are subsets of AI. While AI refers to the broader concept of machines being able to perform tasks that typically require human intelligence, such as reasoning, problem-solving, and decision-making, ML concentrates on learning from data. DL, in turn, is a specialized branch of ML that uses neural networks with many layers to model complex patterns in large datasets [67].

In ML, the learning paradigms differ from each other mainly for the way the learning dataset is built and, in particular, for the presence or absence of labels [68]. The classical taxonomy identifies three broad categories: supervised learning, unsupervised learning, and reinforcement learning [69]. In supervised learning, the algorithm is trained on labeled data, learning to make predictions based on the examples provided [70]. Typical

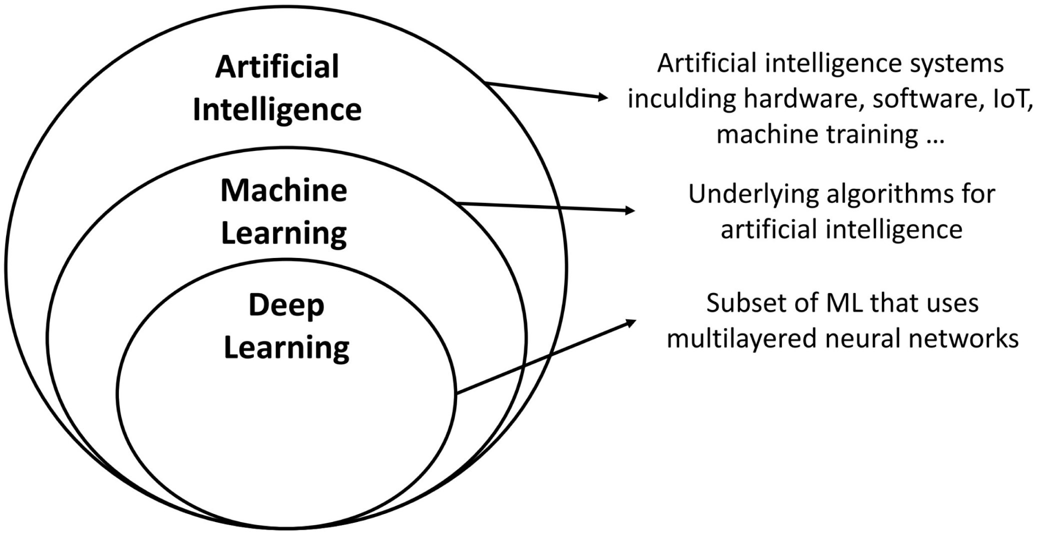


Figure 3.2: AI, ML & DL [67].

applications include tasks such as classification, where each input is associated to one class between two (binary classification) or many (multiclass classification) or to more classes among the ones available (multilabel classification), and regression, where a real and continuous value is predicted [71]. In unsupervised learning, the algorithm analyzes unlabeled data to find intrinsic patterns and structures. Common approaches include clustering, which groups similar items together [72]. Lastly, in reinforcement learning the system learns through trial and error, receiving feedback in the form of rewards or punishments for its actions. In the field of quality inspection and defect detection, the literature emphasizes that supervised learning methods, which are going to be discussed later in the chapter in details, are the most widely used, thanks to the frequent availability of labeled data and their positive impact on quality control. In scenarios with limited data or rare defects, unsupervised approaches and anomaly detection techniques are useful for identifying deviations from normal operation. Reinforcement learning, on the other hand, is still marginal in these contexts and is not a common choice [73].

3.3. Machine learning pipeline for PCB

As described in [74], a typical ML workflow generally includes several phases. First, there is data collection and data cleaning, which is crucial to address typical issues in real-world data such as incompleteness, noisiness, and inconsistency of data. Feature engineering and selection generally follow since not every feature of the dataset may be relevant for the specific use case. The next step concerns model training, which requires the split of the dataset into training, validation and test set. The first two are used for the proper training phase, while the last one is used for the last stage of the process which is the model evaluation. As for this last step, the traditional metrics employed to evaluate a model are accuracy, precision, recall and F1-score, defined as follows [75]:

$$\text{Precision} = \frac{TP}{TP+FP} \qquad \text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

$$\text{Recall} = \frac{TP}{TP+FN} \qquad F_1 = 2\,\frac{\text{Precision}\cdot\text{Recall}}{\text{Precision}+\text{Recall}}$$

Where TP represent the true positives, TN the true negatives, FP the false positives and FN the false negatives.

In the context of PCB inspection, the pipeline typically follows the general steps of the machine learning workflow adapted to the visual domain. The flow generally involves: image acquisition, pre-processing, visual feature extraction, and defect classification/decision.

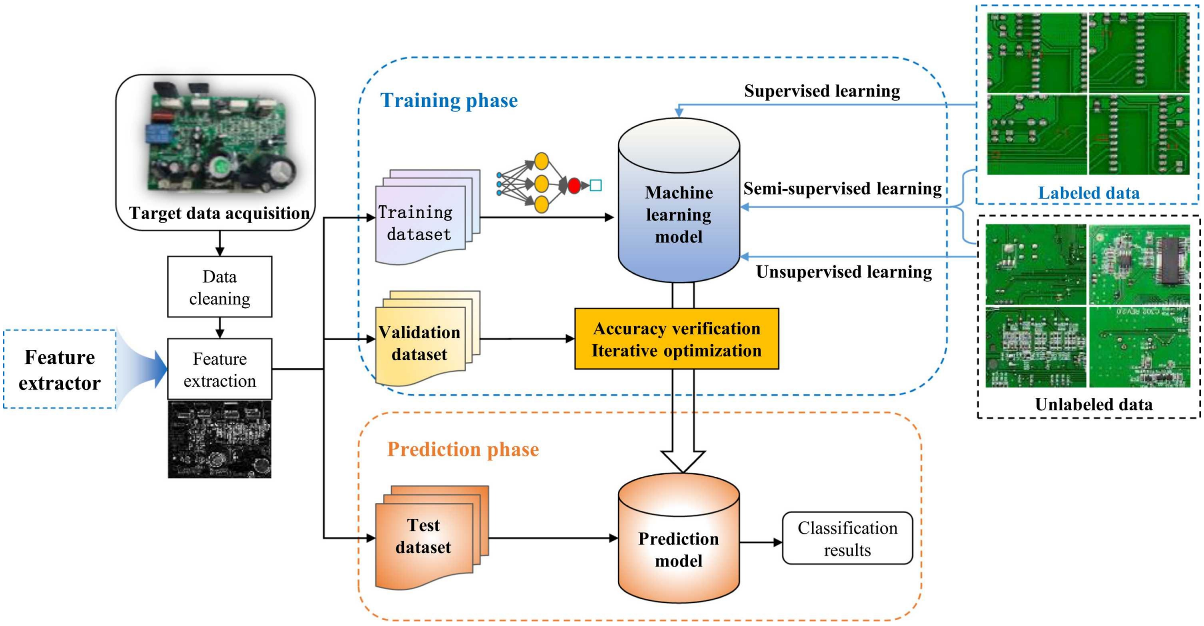


Figure 3.3: PCB defect detection workflow [76].

The starting point is the acquisition of high-resolution images of the PCB whose quality is improved with pre-processing operations including noise filtering (e.g., median or Gaussian filters) and binarization/thresholding to highlight the traces against the background [48]. The following phase, feature extraction, consists in extracting visual features from regions of interest. The beginning of this step are raw pixels that are transformed into a smaller set of descriptors that capture important information such as contours, textures, or shapes present in the image. The goal is to represent the image with a vector of features that can be understood by traditional ML algorithms while reducing noise and data dimensionality

[77]. This process acts as a bridge between unstructured data (images) and the structured inputs required by the models. Among two of the most famous and used feature extraction techniques are Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP). HOG is used to detect shapes and objects and it operates by dividing an image into small cells and counting the distribution of gradient directions (intensity variations) of the pixels for each cell. The result is a histogram for each cell that describes how many edges there are in each main direction. These histograms are then normalized and concatenated into a final feature vector. In simple terms, HOG captures the shape of objects in the image through the orientation of the contours [78, 79]. LBP is a simple method for extracting local textures from an image. For each pixel, it looks at the intensity of the neighbouring pixels and compares it to the value of the central pixel. An 8-bit binary code is constructed indicating, for each neighbour, whether it is lighter (e.g., 1) or darker (0) than the central pixel. Converting this code to decimal gives a "number" that represents the local texture pattern around that pixel. Repeating the operation on all regions (cells) of an image and making a histogram of these codes gives a feature vector that describes the texture of the image. LBP is very fast to calculate and robust to changes in lighting because it is based on relative comparisons between pixels (light/dark patterns) [80–82].

3.4. Employed machine learning techniques

One of the main ML algorithms employed for PCB defect detection is the support vector machine (SVM). A SVM is a supervised ML algorithm that employs a line or, in general, a hyperplane to separate, in principle, two classes, hence it is mainly used for classification problems. The optimal line, or hyperplane, is the one that maximizes the margin, which is the distance between the line, or hyperplane, itself and the support vectors, the extreme points in the dataset, as shown in Figure 3.4. Once the optimal separator has been determined, classifying a new sample simply requires checking on which side of that separator it lies [83, 84]. What if a dataset is not linearly separable in its original dimension? In that case, as shown in Figure 3.5, it can be used a kernel function, which implicitly maps the data from the original n-dimensional space into a higher-dimensional space, where the classes may become separable [84].

In 2014, Hagi et al. [86] proposed an interesting work halfway between a referential approach and a pure ML based one. The study shows how they started from a traditional AOI paradigm based on image subtraction and then moved to automatic classification using SVM to distinguish between real and false defects. In particular, the objective was to distinguish between true defects, such as disconnection, connection, projection and

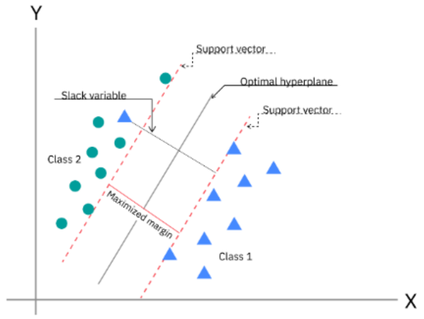


Figure 3.4: Support vector machines [84].

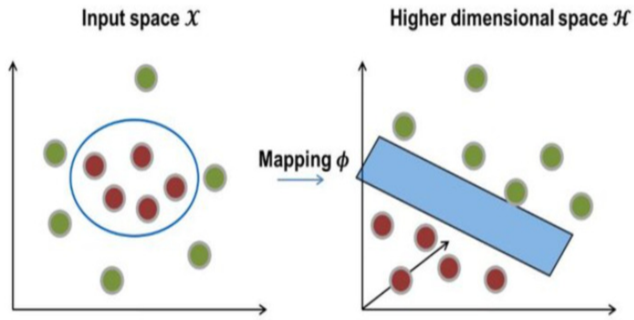


Figure 3.5: Kernel function [85].

cracks, and false defects like oxidation and dust, which can easily generate false alarms in traditional systems. The referential part of the pipeline, where the image of the PCB under inspection is compared with a golden sample, is used to find and highlight areas that might correspond to a defect. For each identified region, both color and shape characteristics (such as area, perimeter, and roundness) are extracted. The classification is dealt with using an ensemble of SVMs, each trained on balanced and random subsets of the dataset, that are combined by weighted majority voting: if many SVMs agree that an anomaly is a true defect, the final classification will follow this indication; Otherwise, it will be labeled as a pseudo-defect. The weights serve to give greater importance to the most reliable classifiers, improving the robustness of the overall system.

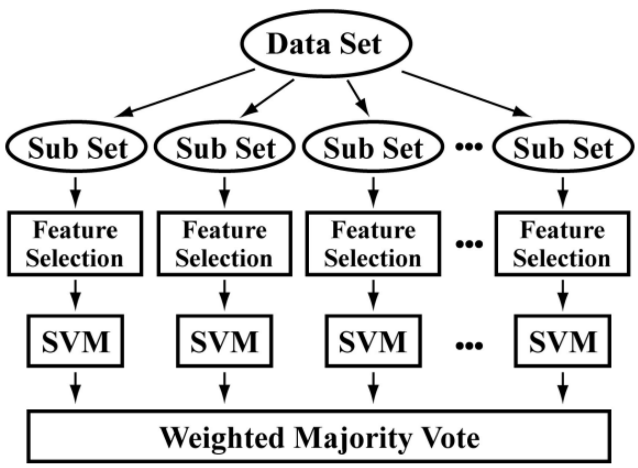


Figure 3.6: Ensemble SVM [86].

Tested on an industrial dataset with 634 defects, the proposed method achieves 93.45% accuracy on color images, outperforming both a fixed threshold approach and a single

SVM. Moreover, it reduces false negatives on true defects and introduces a more flexible and adaptive classification thanks to machine learning.

Similarly to the approach proposed by Hagi et al. [86], Chen et al. [87] proposed a system that uses the referential stage to detect where anomalies may be present, but relies on SVM classifiers to decide whether they correspond to actual defects, such as open, short, pinhole, under-etch, and mousebite. Compared to a traditional rule-based AOI method based on XOR and handcrafted thresholds, this system achieves both higher accuracy and significantly reduced processing time (124 ms vs. 530 ms), demonstrating the practical advantages of this approach.

Another hybrid method combining a traditional referential stage—based on image subtraction and an SVM classifier for defect recognition was proposed by Zhang et al. [88]. After detecting candidate regions, a set of grayscale and geometric features is extracted and used to train a single optimized SVM via grid search. The system reached 96.67% accuracy on a test dataset of 60 images, confirming the effectiveness of integrating machine learning in the decision stage.

Differently from the previous works that required a referential approach as first step, Wu et al. [89] proposed a method entirely based on ML to classify solder joint defects, including good solder, pseudo solder, solder insufficient, component shifted, wrong component and tombstone. Solder joint images are acquired and, from each of them, simple features are extracted and used to feed a two-stage classification pipeline. In the first stage, a Bayesian classifier determines whether the joint is compliant; if it is not, a multi-class (one-vs-one) SVM assigns it to one of the predefined defect classes.

Model	Dataset	Main results	References
Referential stage & Ensemble of SVM classifiers	Industrial PCB dataset: 634 defect samples (203 true defects: disconnection, connection, projection, crack; 431 pseudo defects: oxidation, dust) from color board images	Achieves up to 93.45% accuracy on color images, outperforming a fixed-threshold referential method and a single SVM, and reducing mis-classification of true defect	[86]
Referential stage & SVM classifier	Ad-hoc industrial PCB dataset: 400 color images with five common pattern defects (open, short, pinhole, underetch, mousebite; 80 images per class) from real or simulated faulty boards, plus additional boards with cosmetic defects (skip plating, copper exposure, oxidation)	Achieves 100% detection rate for the five pattern defects and 93% SVM classification accuracy, while reducing total inspection time to about 124 ms per image (vs 530 ms for a rule-based XOR method)	[87]
Bayesian classifier & SVM	Ad-hoc color solder joint dataset: 280 chip components (type 0402) imaged with RGB dome illumination and CCD color camera; 6 classes (109 good, 48 pseudo, 30 insufficient, 44 shifted, 30 wrong component, 19 tombstone)	With seven selected features, the two-stage Bayes+SVM scheme achieves 100% classification accuracy and outperforms k-NN, decision tree and BP networks while using fewer features and shorter modeling time	[89]

Table 3.1: Summary of SVM approaches for PCB defect detection.

Another supervised learning classifier discussed and used in the defect detection literature is decision trees (DT). The name of this classifier comes from the tree structure representing a logical sequence of binary or multi-value decisions, constructed from input data (features) to predict a target variable. The construction of the tree starts from the root node which sets a first discriminating condition on the dataset. Each answer corresponds to a branch that divides the data into subsets, giving rise to new decision nodes. This process continues recursively until the leaf nodes, representing the final decisions, hence the final class the sample under exam is assigned, are reached [90]. The main advantage of this method is that it represents one of the most interpretable ML models, as shown in Figure 3.7.

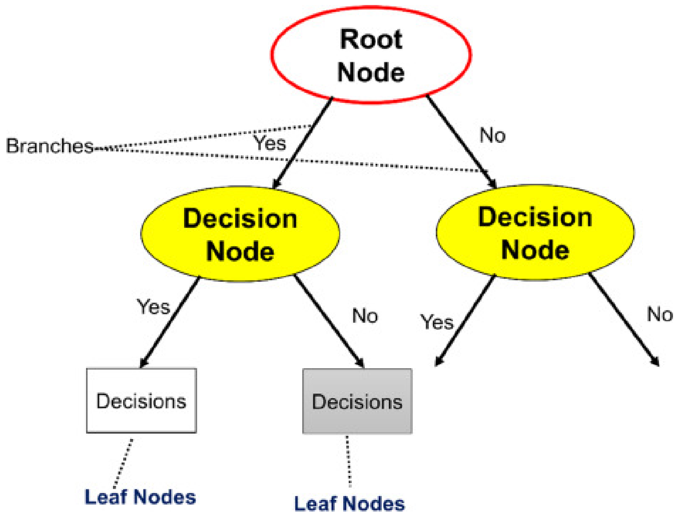


Figure 3.7: Decision tree [91].

In 2002, Jiang et al. [92] employed a DT to classify PCB golden finger defects, in particular scuffing, blotted tin, exposed nickel and unplating. More in detail, it is described a logistic regression tree (LRT) where each node in the tree does not perform a simple thresholding, but applies logistic regression to separate classes based on linear combinations of RGB values. The tree is built in a hierarchical manner and, at each step, the defects are separated in pairs (e.g. scuffing vs others, then unplating vs. others ecc.) and the split conditions that maximize the distinction between the groups are automatically learned. This approach, which is entirely based on the raw RGB colours without the need to extract complex features, achieved an accuracy of 89.83%, overcoming other supervised ML models such as Bayes classifier.

Xi et al. [93] proposed a two-step classifier involving DT to classify solder joints, in particular good solder (GS), missing (MS), shift (Sf), tombstone (TS), surplus solder (SS), lack solder (LS) and pseudo joint (PJ), reported in Figure 3.9. Each acquired image

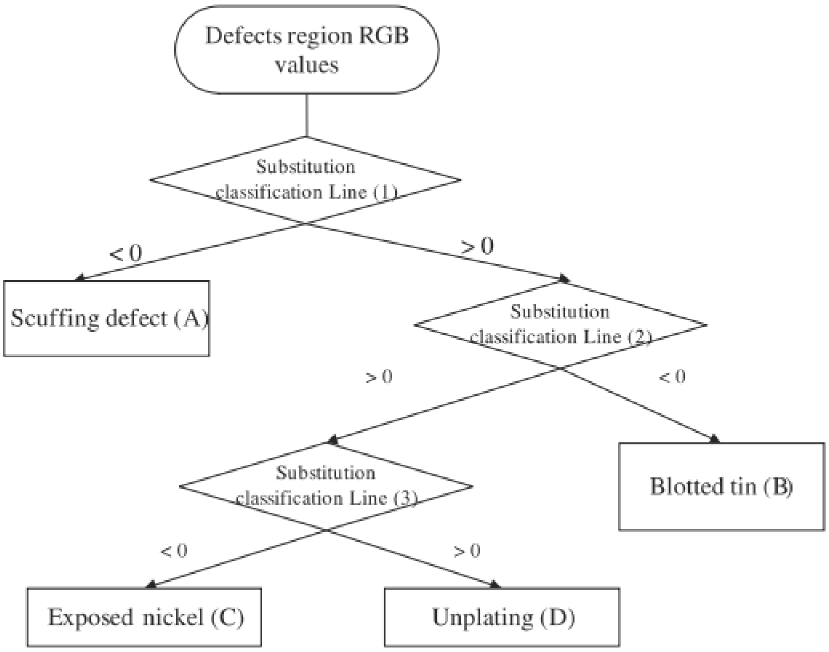


Figure 3.8: Logistic regression tree [92].

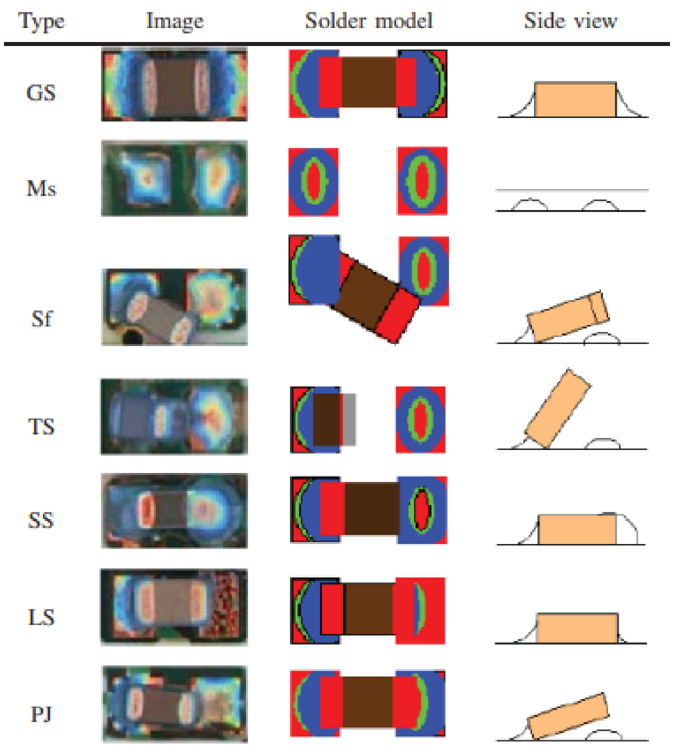


Figure 3.9: Solder joint defects [93].

is divided into sub-regions and, for each, a small but informative number of features is extracted. Then, each sub-region is classified with an AdaBoost classifier that provides a binary result, either "ok" or "ng". These local results are combined into a few patterns (S1-S6) and represent the input for the final DT which provides the final classification. The proposed classifier achieved a recognition rate of 97.2% which is a significant improvement with respect to the 93.1% of traditional image comparison methods.

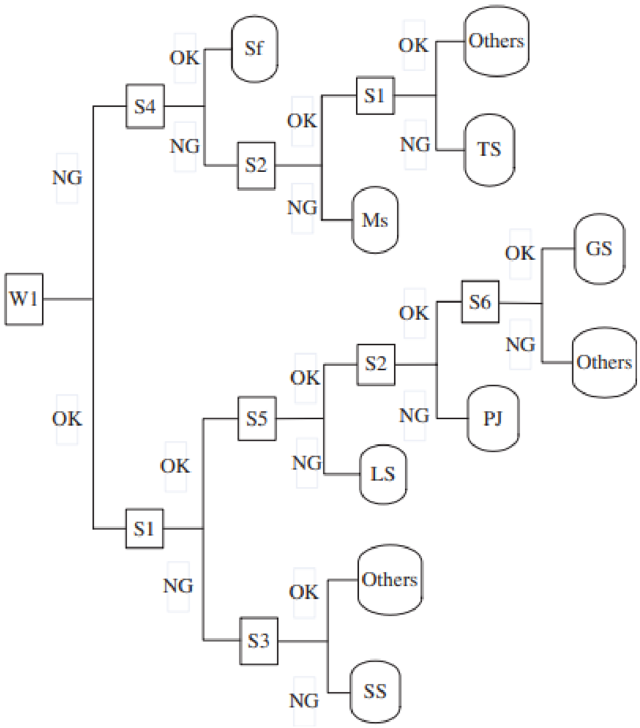


Figure 3.10: Decision tree classifier [93].

Despite all its advantages, a DT classifier also comes with downsides, the main one being its tendency towards overfitting: the deeper the tree, the higher the chance it overfits. In addition, it is a really unstable model, which means that little variations in the training set can lead to very different trees.

To overcome these limitations, the random forest (RF) model has been conceived. RF is an ensemble method which can be thought of as a forest of DT. The key idea is to fit several DT classifiers on different sub-samples of the dataset and combine their output: a majority vote among trees, for classification problems, or an average, for regression ones [94], as shown in Figure 3.11.

Yuk et al. [96] proposed a method to detect scratches and improper etching that employs the RF model. The starting point is the acquisition of the PCB image and, using the Speeded Up Robust Features (SURF) algorithm, special points (corners, changes in bright-

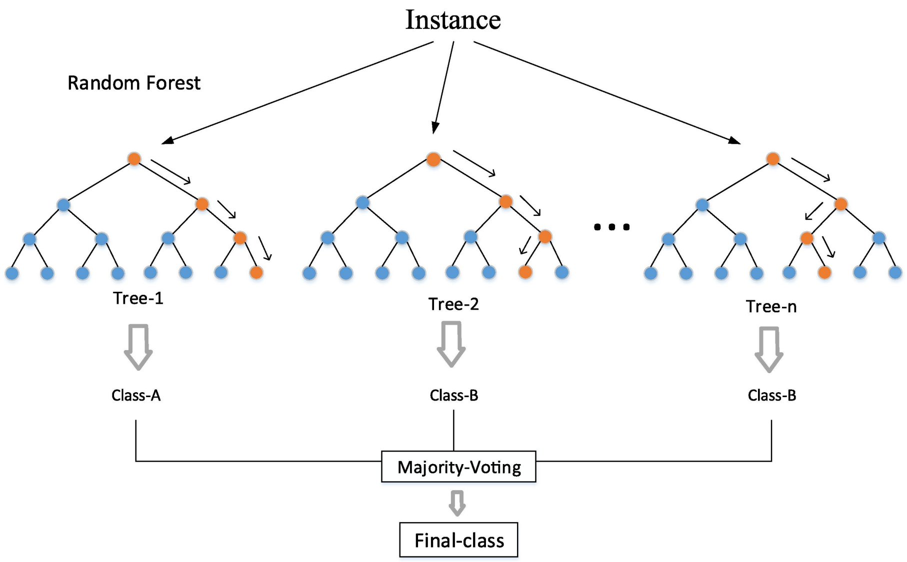


Figure 3.11: Random forest [95].

ness, distinctive points) are found. For each of those points, a descriptor is built, which is a set of numbers that describes what happens around that point. These descriptors represent the input vectors for the RF model. Each tree looks at various characteristics of the point of interest and decides whether that point appears to be defective (part of the scratch) or defect-free. Ultimately, the RF assigns each point a probability of being a defect. After that, it is built a Weighted Kernel Density Estimation (WKDE) which allows to see where clusters of points with high probability are located: a scratch, for example, is likely to generate many "probably defective" points close to each other, and this is useful since it allows noise to be separated from true defects: a single point with a high probability, but isolated, may be an error or a false alarm. If, on the other hand, many nearby points are labeled as defective, it is more likely to be a real defect.

Model	Dataset	Main results	References
Logistic regression tree classifier	Ad-hoc color PCB golden finger dataset: 4 defect types (scuffing, blotted tin, exposed nickel, unplating); images acquired at 512×512 with a CCD camera and LED lighting; 2000 sampled defect regions for training and 1200 defect images (300 per class) for testing	chieves 89.8% average correct classification rate, outperforming Bayes, LDF, minimum distance and K-NN, while relying only on RGB values without complex feature extraction	[92]
Sub-region classification with AdaBoost, followed by a decision tree for final multi-class solder joint diagnosis	Ad-hoc non-public dataset built from 0402 chip solder joints imaged with a CCD camera under RGB dome illumination; color patches of size 42×118 px covering 7 classes (good, missing, shift, tombstone, surplus, lack, pseudo joints).	Cross-validation recognition rate 97% (97.2% online) versus 93.1% for image comparison, with competitive inspection time (8.6 ms), showing a good accuracy–speed trade-off for industrial AOI.	[93]
SURF feature points classified by a Random Forest	Ad-hoc non-public set of 10 high-resolution color PCB images with thin scratch faults from a battery manufacturer;	Mean AUROC 0.91, clearly better than RF or KDE alone, and can detect scratches without any reference image, robust to size, rotation and position changes	[96]

Table 3.2: Summary of DT & RF approaches for PCB defect detection.

Among the other traditional ML algorithms that can be used, we can cite k-nearest neighbors (knn). Knn is a supervised learning classifier that uses proximity and the concept of similarity to make classifications or predictions. The core idea to classify a new sample is to select the k points in the training set that are "closest" to it and assign the new sample the most frequent class among those of the k selected points [97].

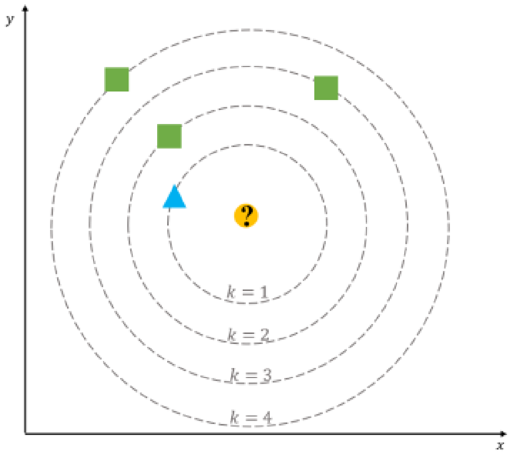


Figure 3.12: Knn algorithm [30].

The key difference between this technique and the ones described previously is that knn is part of the so-called *lazy learning* models, which means that there is actually no model to train, such as a straight line in linear regression or a tree for decision trees. It only stores a training dataset, and the model is the data themselves. This also means that computation takes place entirely at the moment of classification or prediction [98]. Liu et al [99] proposed an improved knn to detect solder joint defects. The system works on images captured by an industrial camera: for each joint, a region of interest (ROI) is extracted and geometric features such as area, circumference, roundness, and intensity variance are calculated. Those features represent the input vector for the knn classifier which, as described before, classify each joint based on the k closest examples in the dataset. This method, used to address defects such as more solder, less solder, leakage welding and connection welding, proved to be more efficient with respect to traditional AOI systems. The results, obtained with 90 training solder joints and 209 testing solder joints across the different categories, are reported in the Table 3.3 below.

Detection method	Type	Precision rate
The improved knn method (k=3)	More solder	96.875%
	Less solder	100%
	Normal solder	99.286%
	Comprehensive	99.043%
Knn method (k=3)	More solder	90.625%
	Less solder	94.595%
	Normal solder	97.857%
	Comprehensive	96.172%
AOI system	More solder	93.75%
	Less solder	94.595%
	Normal solder	96.428%
	Comprehensive	95.694%

Table 3.3: Comparison of precision rates by method [99].

3.5. Advantages and limits of machine learning techniques

Based on the previous analyzed papers and beside the specific ML model employed, we can highlight different common advantages of this approach. First, a benefit of this data-driven approach is the reduction of false alarms [86]: models trained on real data are better at distinguishing true defects from acceptable variations, resulting in lower false detection rates compared to conventional optical inspections. In addition, the main advantage is the possibility to overcome the performance of traditional 2D AOI systems. All the analyzed works demonstrate not only high levels of accuracy, precision, and detection rate on their own, but also a clear performance improvement when directly compared with traditional AOI systems. A significant improvement in performance also concerns the inspection time, which in most cases is noticeably shorter than that of traditional AOI systems. This reduction in processing time has been clearly demonstrated, for instance, by [87], where the proposed approach achieved faster inspections while maintaining high levels of accuracy. Moreover, another key advantage is the absence of any requirement for a golden sample or reference image during the inspection process. While a few of the analyzed works [86–88] opted for a hybrid strategy, in which an initial referential step is followed by a ML-based classification stage, the majority relied on a fully ML-driven pipeline. In these cases, the defect detection and classification are carried out entirely through learned models, eliminating the dependence on referential steps, hence avoiding the constraints typically associated with template-based methods.

Despite bringing many advantages, ML approaches also come with the general downsides of ML techniques. First, the model's performance depends heavily on the availability of a large and representative training dataset: if defect data is scarce or unbalanced, especially for rare anomalies, the classifier may struggle to recognize them reliably. In other words, the system's "knowledge" is linked to the defects seen during training [100]. Directly linked to that, the results of these techniques heavily depend on the quality of the extracted features: classical (non–deep) ML pipelines typically require manually defined descriptors (e.g., shape, color, histograms) to be provided to the classifier, and performance may deteriorate if the selected characteristics do not effectively capture all defects. Furthermore, some algorithms have inherent critical issues: for example, knn is sensitive to noise and outliers, and the result may vary depending on the choice of the optimal k hyperparameter [101]. Finally, even though some models (e.g., DT) are easily interpretable, others often used in practice, such as SVMs or Random Forests, are frequently regarded as black boxes, which can make it harder to explain a specific decision after it is taken [102, 103].

4 | Deep Learning techniques for PCB defect detection

As already said in the previous chapter, DL is a specialized branch of ML that uses neural networks (NN) with many layers to model complex patterns in large datasets. At a high level, a NN is a human-brain-inspired computational model which is based on the cooperation of neurons, also called perceptrons. Neurons are the fundamental and basic processing units of NN and are arranged in layers (input layer, one or more hidden layers, output layer) and linked by weighted connections, simply called weights, that are the parameters whose values are adjusted during the training phase [104].

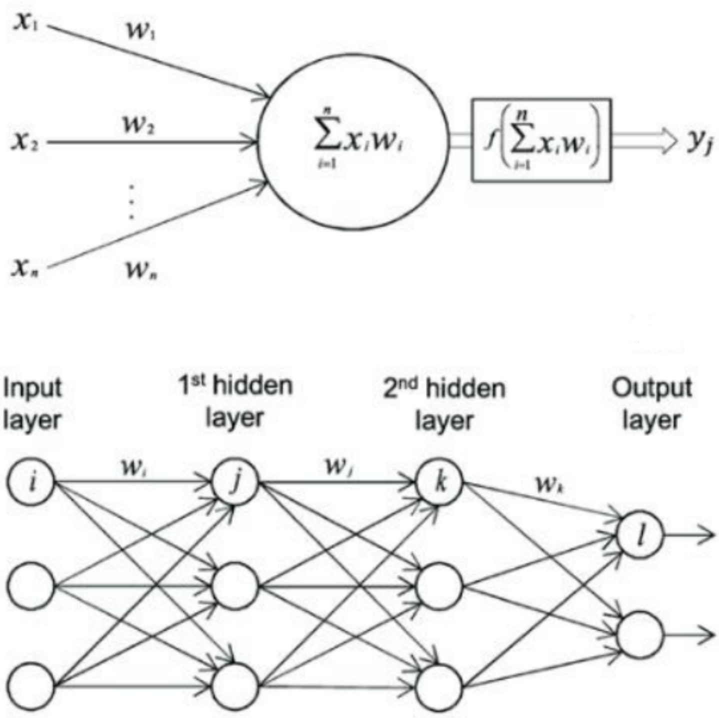


Figure 4.1: Neuron (up) and neural network (down) examples [105].

Being a direct subset of ML, DL can be employed to address both supervised tasks (e.g., classification, regression) and unsupervised ones (e.g., clustering or feature extraction)

[106] and its models can be evaluated using many of the same metrics commonly adopted in traditional ML, such as accuracy, precision, recall, and F1-score, beside additional metrics (e.g. intersection-over-union in image segmentation tasks). The main difference between ML and DL is that, in the former, the features to be used and learned by the models must be manually defined based on the specific problem to solve, which requires a significant effort in feature engineering, whereas in the latter, features are automatically learned from data during training.

4.1. The dataset role and construction in deep learning pipelines

One of the main consequences of this distinction is that DL models typically require much larger datasets than traditional ML approaches, as the automatic extraction of complex features and patterns demands more data. Producing a high-quality dataset is a crucial step to enable an efficient defect detection and it requires both gathering an important amount of PCB images and labeling them accurately, indicating the presence of both the different components and the localization of defects.

The first step in the dataset construction is represented by the gathering image step, which can concern real data and artificially generated data. In the industrial setting, images can directly be taken from AOI systems. For example, in a recent study 32,259 images have been taken from the quality management system of a PCB factory, using high-resolution linear scanners integrated into AOI machines for the DsPCBSD+ dataset [107].

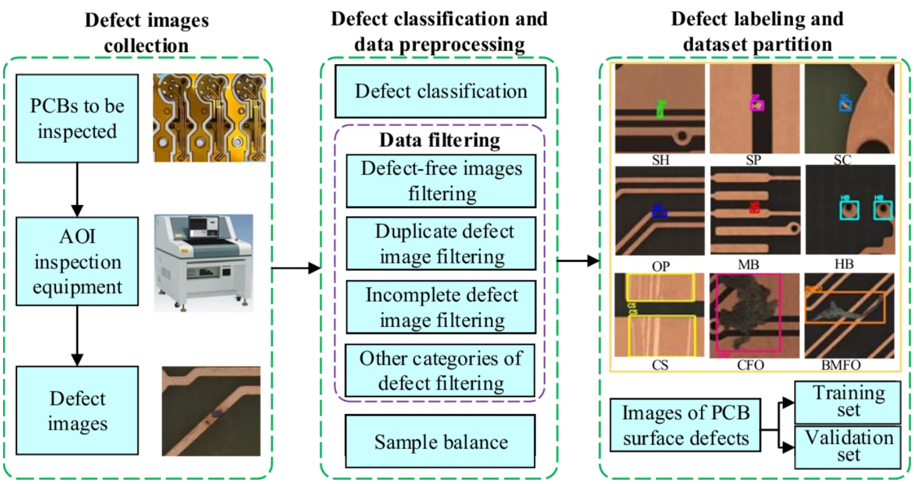


Figure 4.2: Overview of the construction process for DsPCBSD+ [107].

In the academic and research domain dedicated acquisition systems have been developed as well. The SolDef_AI dataset, for example, has been built using an artificial vision platform that captures each component from three angles (from above, at 45°, and axonometric view) under different lighting conditions, to ensure a complete view of the welded joints [108]. Eventhough this acquisition process may appear easy to implement, it actually shows different limitations. First, defects are rare and acquiring thousand of images of a certain defect can be difficult. In addition, companies are reluctant to share images of their proprietary PCB designs. Given that, a lot of academic dataset are either created by artificially inducing defects or employ extensively data augmentation, which refers to the generation of different samples starting from the same image. For example, a single image can be rotated and then cropped, providing two more and different images to the model. This allows to only gather a relatively small number of images and rely on this technique to complete the dataset. Example of that are the DeepPCB [109] and PKU-Market-PCB datasets which comprise, respectively, 1,500 and 1,386 defect images covering missing hole, mouse bite, open, short, spur, pinhole and spurious copper. In the first case, defects have been artificially added to the defect-free samples resulting in images with 3 to 12 defects, while to the PKU-Market-PCB dataset an extensive data augmentation pipeline has been applied and brought the dataset up to 10,668 images using geometric and image transformations [107]. Also private datasets have been developed. Hu et al. [110] built a dataset of originally 1,500 defect images, which then reached 12,000 samples thanks to data augmentation, including, in addition to the just mentioned defects, also solder balls. Liao et al. [111] developed a dataset including 19,029 defect images, 2,008 original and the rest augmented ones, spanning different defects such as bumpy, clutter, scratch, line repair damage, hole loss, or over oil-filling.

For a dataset to be truly useful for training DL models, it is essential that the annotations are consistent with the type of task to be tackled. The construction of these datasets involves careful labeling, which often turns into the bottleneck in AOI project because transforming raw images into training data requires costly, human and prone to error intervention. The literature shows mainly 4 labeling techniques: image-level, bounding boxes, segmentation and keypoint annotation. The simplest labeling technique is the image-level annotation, in which an image is simply assigned an OK/NG label, without indicating where the defect is located. Obviously, labeling each image as defective or not is quick but the clear limitation is the total absence of spatial localization of the defect in the image, making this labeling technique only suitable for simple classification tasks [112]. The bounding box (BBox) technique plans to enclose each defect in a rectangle defined by coordinates, generally x,y,w,h. It is the most used labeling technique in PCB defect detection dataset for objection detection and it is supported by numerous studies, for example the PCB-Defect dataset gathers 230 images with 1,704 defects annotated with BBox and memorized in the COCO (Common Objects in Context) JSON format with precise coordinates by category and location [113]. Figure 4.3 shows an example of this labeling technique.

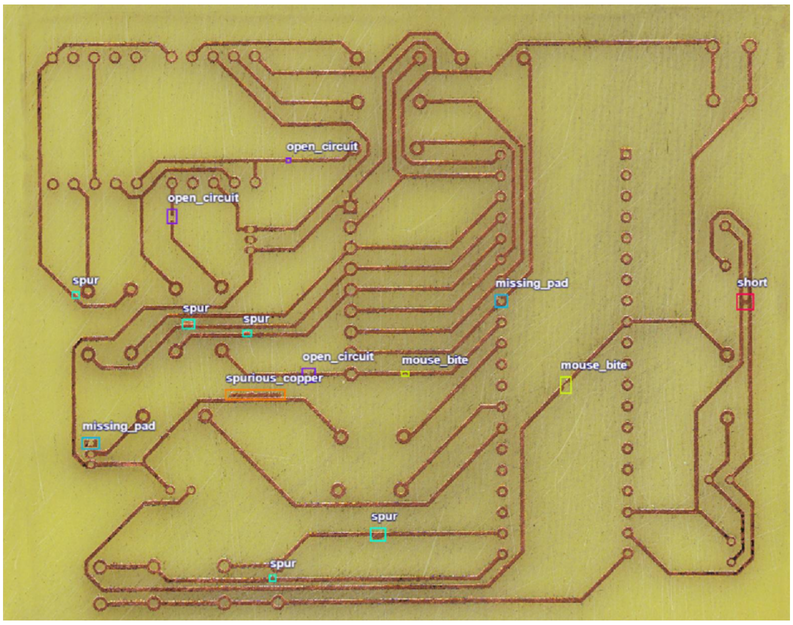


Figure 4.3: BBox labeling example [113].

Segmentation technique provides an exact outline of the defect in the form of a pixel mask or polygon, offering greater granularity than BBoxes. For example, the MeiweiPCB dataset [114] provide both bounding boxes and masks as shown in Figure 4.4. Mineo et al. [115] built the PCB-SAID dataset with a two-stage labeling pipeline: first BBox for localization and then polygonal masks for a more accurate outline.

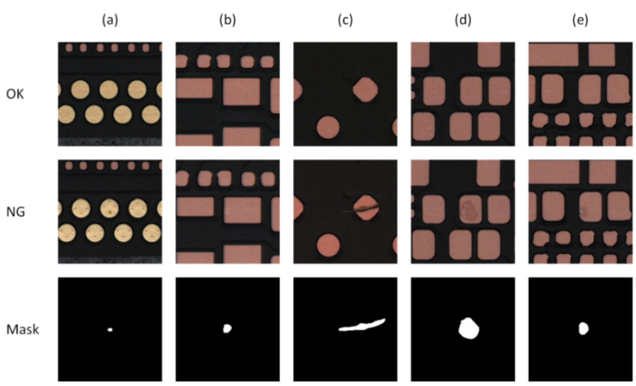


Figure 4.4: Segmentation labeling example [114].

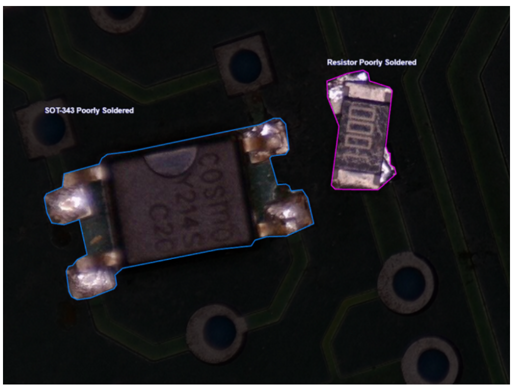


Figure 4.5: Polygonal masks example [115].

This approach makes it possible to obtain extremely accurate annotations, which are essential for complex defects, but it requires much more time and expertise. The keypoint labeling technique consists of marking specific coordinates (points) associated to defects or components instead of identifying the whole area, which can be way faster and significantly reduces the effort of manual labeling. With respect to the previously mentioned techniques, it is less employed but it still find place, for example Chung et al. [116] propose a keypoint-based inspection system: each component is supervised via a keypoint in its centroid and an estimate of its size (width/height), avoiding the use of anchor boxes and allowing the position and size of the component to be derived.

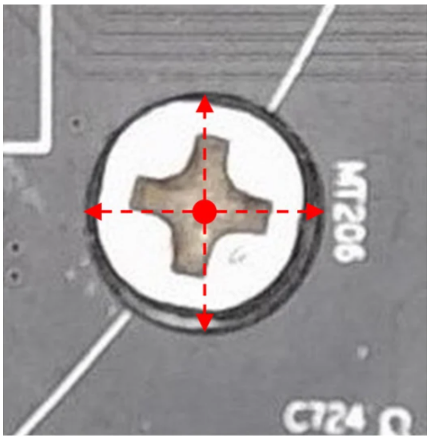


Figure 4.6: Keypoint labeling example [116].

4.2. Image and patch classification

Convolutional neural networks (CNN) are NN specifically developed to deal with images as input and to solve tasks such as image classification, semantic segmentation, object localization and detection, facial recognition, image generation and others. A CNN for classification is composed of two main parts [117, 118]:

- Feature extraction network (FEN): It represents the first part of the architecture and it is responsible for taking the image as input, extracting the key features, through the convolution operation between the filters, also called kernels, and the pixels in the image, and producing the latent representation vector. It is also called convolutional block;
- Classifier NN: It takes as input the latent representation vector produced by the FEN and perform the classification.

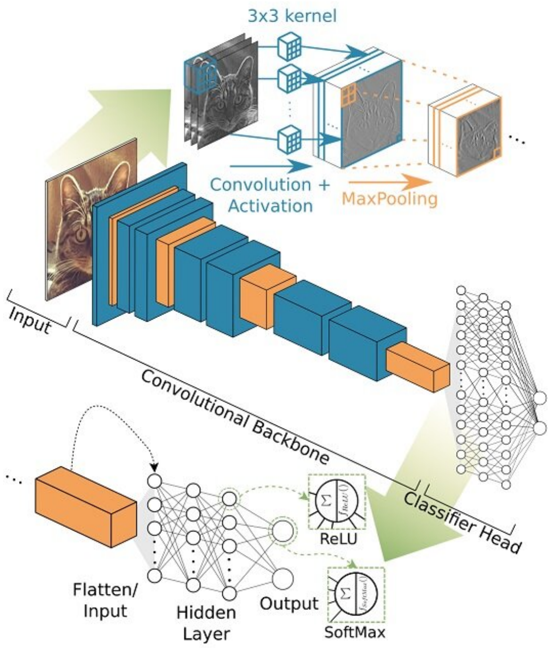


Figure 4.7: CNN architecture [119].

Given the fact that they were designed to deal with images, CNN and derived architectures represent the main tool used in deep learning approaches for PCB defect detection. In the PCB Welding Defects data pipeline by Martiri et al. [120], a classical CNN plays a central role in classifying defects using images of individual PCB components that have been pre-localized via an automated inspection pipeline. The authors leverage AOI output images (each containing multiple components) together with AOI result logs (CSV) and PCB design diagrams (PDF) to define precise bounding boxes for each component, extracting standardized single-component images with normalized lighting to reduce illumination variability. The CNN architecture itself is relatively compact: it consists of three sequential convolutional blocks (each convolution layer followed by max-pooling) for feature extraction, and two dense (fully-connected) layers for the classification output. This network is employed for both binary classification (PASS vs. FAIL for a component) and multi-class classification, identifying the specific defect type when a fault is present. In the multi-class mode the final layer was adjusted to output 13 classes (12 defect categories plus one non-defective class) to cover all defect types in the dataset. Notably, although a classical RF was also implemented for comparison, the CNN demonstrated superior robustness on noisy or degraded images. For example, under significant brightness variation (when test images alone had altered brightness), the CNN maintained high accuracy (83% vs. 70% for the RF).

Even if classical CNN, as just described, can be employed to perform classification tasks, the current literature shows how two main classes of algorithms and architectures emerged within the scope of PCB defect detection and classification: two-stage algorithms and single-stage algorithms.

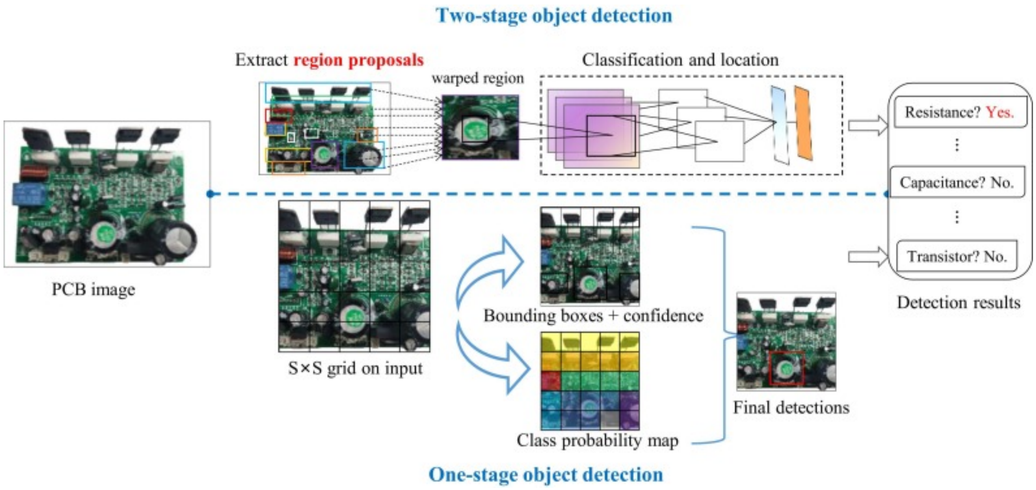


Figure 4.8: Two-stage vs one-stage architecture [37].

Two-stage algorithms divide the process into two sequential steps:

- A network identifies probable regions of interest (region proposals) containing possible defects;
- A classifier (often a CNN model) analyzes each cropped region to confirm the presence of the defect and determine its class.

Classic examples of this paradigm are the Region-CNN (R-CNN), Fast R-CNN and Faster R-CNN, which introduces an integrated Region Proposal Network to generate proposals in a learnable way. This two-stage strategy tends to be very accurate at locating minute defects and distinguishing subtly different classes because the proposal stage can focus on local details and small objects, but it involves greater computational complexity and latency, making them difficult to operate in real time [121].

Beside two-stage detectors, they also emerged, and established as state of the art, one-stage detectors that perform both localization and classification of defects in a single pass. The progenitor of this family is the You Only Look Once (YOLO) architecture which is based on the idea of reformulating detection as a direct regression problem: given the entire image, the network simultaneously predicts the coordinates of the bounding boxes and the categories of objects (defects) present, without an explicit region proposal phase. This makes the process extremely fast and suitable for real-time applications. In PCB contexts, YOLO methods have been rapidly adopted because they allow optical inspection to be performed in-line during production at very high speeds. The main advantage of one-stage approaches is therefore efficiency: the simplified pipeline avoids redundancies and can be optimized for dedicated hardware. In addition, many recent one-stage models are lightweight (low complexity) and well suited for deployment on embedded and edge systems directly on the production line. Compared to Faster R-CNN, a YOLO model typically takes up less memory and performs fewer calculations per image, while maintaining good accuracy [122, 123]. Indeed, a recent comparative study [124] on PCB inspection explicitly benchmarks Faster R-CNN against modern YOLO detectors for PCB component localization and classification, showing that two-stage approaches can be penalized in this domain by both accuracy and speed constraints. In that work, Faster R-CNN variants are evaluated together with YOLOv8 and YOLO11 on structured PCB datasets, and the authors report that one-stage detectors consistently provide a better trade-off between detection performance and inference time for high-throughput AOI settings. Beyond confirming the advantage of one-stage detectors, Martiri et al. [124] provide a concrete example of how recent YOLO versions can be tailored to PCB inspection requirements. Their best-performing configuration is a YOLO11 medium model equipped with Oriented Bounding Boxes (OBB), which is explicitly motivated by the presence of non-axis-aligned components and rotated board regions in real acquisition setups. Under a multi-class setting (seven component types plus a dedicated MISS class for missing components), the YOLO11m-OBB variant achieves up to 99.0% mAP@50 and 97.1% F1-score for multi-class detection, and 98.8% F1-score specifically for missing component identification. The same study also evaluates resilience to degraded images by varying brightness and introducing blur. Results indicate that the selected YOLO11m-OBB model remains stable across broad luminance and focus variations, with sharp performance drops only under extreme degradation—supporting its suitability for less controlled acquisition conditions.

In this technology advancement, a key turning point was the evolution from YOLOv5 to YOLOv8, which brought significant benefits to PCB inspection. Indeed, YOLOv5 (2020-2022) was based on predefined anchor boxes and showed limitations in detecting very small objects (minute defects) due to the difficult calibration of anchors. YOLOv8 (2023) has solved many structural limitations. The key difference is an anchor-free architecture: the model directly predicts the normalized coordinates of the boxes from the feature map points. This eliminates the need to manually design and calibrate anchors and improves generalization on datasets with very different shape statistics, which, in the context of PCBs, is advantageous because defects have highly variable shapes and aspect ratios (e.g., a long, thin scratch vs. a circular missing hole) [125, 126]. In addition, two strategies that have emerged in the literature are the use of lightweight backbones and efficient convolution operators:

- Ghost convolutions: This convolution module is based on the observation that many feature maps in CNNs are redundant and it generates part of the feature maps with normal conv2D and the rest through simple linear transformations on the first maps. By integrating GhostConv modules into YOLO (e.g., in variants called Ghost-YOLOv8), a 50% reduction in parameters and FLOPs was achieved compared to standard convs, without significantly degrading the accuracy in extracting relevant PCB defect features [125, 127]. Similarly, Ji et al. [128] built a light version of YOLOv8 called G-YOLOv8, where standard convolutions are replaced by GhostConv blocks. On the DeepPCB and PKU PCB datasets it runs at about 125 FPS and even slightly improves MAP compared to the original YOLOv8, showing that it is possible to push real-time speed very high without losing detection accuracy.
- Backbone MobileNet: Other studies have replaced YOLO's backbone with lightweight architectures such as MobileNetV3. For example, Wu et al. [129] created a YOLOv4-MobileNet hybrid that reduced the model weight improving MAP and inference speed [125].

Other important innovations that brought the YOLO architecture to be the state of art concern new attention mechanisms:

- Convolutional Block Attention Module (CBAM): This module applies channel attention and spatial attention in sequence. Channel attention teaches the network what to look at, identifying which feature maps contain useful information (e.g., those highlighting contrast differences in copper or anomalous edges). Spatial attention, on the other hand, indicates where to look, emphasizing regions in the image where a defect is likely to be present. Integrating CBAM into YOLO's downsampling blocks has shown benefits in keeping track of small defects [125];
- Simple Attention Module (SimAM): It is a more recent attention mechanism, noteworthy because it does not introduce additional parameters. It exploits an energy theory of neurons: adjacent pixels with similar values receive higher weights (assuming they belong to the same structure), while isolated or anomalous ones are attenuated. Although simple, SimAM improves the network's ability to distinguish defects from complex backgrounds [130]. For example, Zhang et al. [131] proposed LPCB-YOLO, a lightweight version of YOLOv8 designed for PCBs which employs GhostConv together with self-similarity weights inspired by SimAM. In practice, the model, which is actually smaller than YOLOv8, learns to give more importance

to small, truly informative regions and to better ignore the background without increasing the model size. Tests on the industrial HDPCB dataset and on the public DeepPCB dataset show that LPCB-YOLO maintains (or slightly improves) precision and recall compared to YOLOv8, while requiring fewer parameters and fewer computations, which is useful for detecting subtle defects even on resource-limited hardware.

4.3. Semantic segmentation for PCB inspection

Unlike the classical image classification task where a label is assigned to the whole image, in a semantic segmentation task, a label is assigned to each pixel of an image, increasing the granularity of the inspection. In this field, the U-Net architecture, and its derivations, still represents the state of the art and it consists of two main parts: an encoder and a decoder. The first downsamples the image through a series of convolutional layers like in a classical CNN, while the second builds back the image features using deconvolutions [132].

Among the first attempts to introduce semantic segmentation into PCB defect detection, they can be highlighted hybrid models that combined deep learning and referential techniques. A representative example is the one carried out by Ling et al. [133] that proposed a Deep Siamese Semantic Segmentation Network for the detection of soldering defects, including missing solder, missing part, solder short, overturned direction and solder ball. The proposed network takes two images as input, the one being examined and the reference image, and uses a Siamese architecture to calculate the semantic differences, highlighting the defective regions through a subsequent pixel-level segmentation phase as visible, for example, in Figure 4.9 which shows, from left to right, input image to be checked, template image, ground-truth mask and the detection mask.

As already seen with ML techniques, research has gradually moved from hybrid methods towards end-to-end networks capable of operating on a single input image, without any reference image. Li et al. [134] proposed a method based on an improved U-Net, in which the classic VGG16 encoder is enhanced with a Dynamic Hybrid Attention Module to handle background noise, which improved accuracy and mean intersection over union with real-time processing at approximately 30 fps.

A pivotal work for the development of semantic segmentation in the PCB context is represented by the one published in 2024 by Li et al. [135] in which defects such as solder bridge, open connections, pin holes, pin missing and component missing are investigated. The method employes semantic segmentation with rule-based analysis based on the PCB

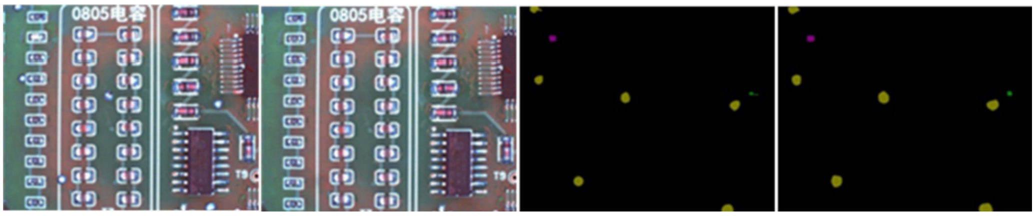


Figure 4.9: Detection results [133].

design diagram. In practice, as shown in Figure 4.10, the system uses a lightweight segmentation neural network to label each pixel in the board image as belonging to certain categories. This step generates a segmentation map of the entire board, highlighting in different colors the regions corresponding to components. The system compares the obtained feature map with an ideal map generated by the CAD design of the circuit. The two images are aligned using reference points and analyzed with a rule-based algorithm, which checks where elements are missing or where anomalies appear.

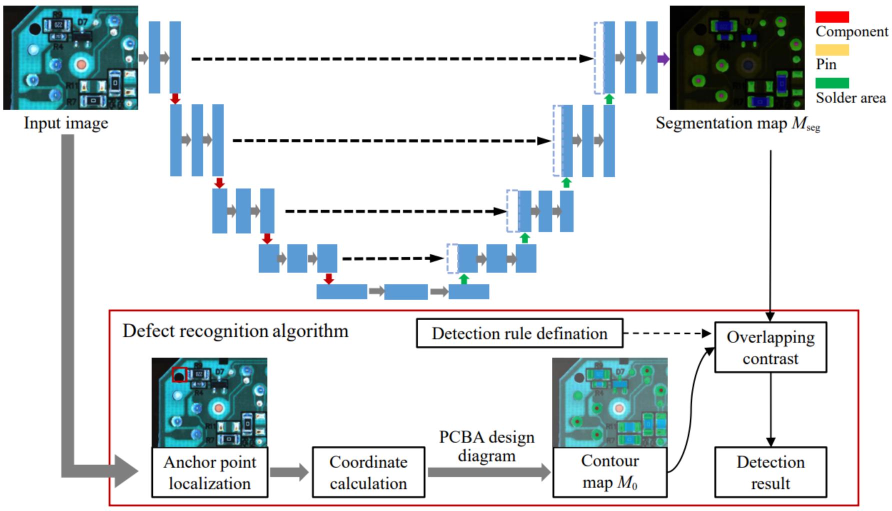


Figure 4.10: Adopted method in [135].

Among the novelties of the adopted approach described in the paper [135], one of the key points consists in the intention to implement this solution on lightweight computing devices (Edge Intelligent System) rather than industrial PCs. To make that possible, different improvements have been introduced, including group convolution and quantization. Compared to conventional image comparison-based inspection algorithm, such as XOR method on a "golden template" image, the experimental results, collected on real production lines with boards containing known defects, show very significant improvements over traditional AOI methods:

- Accuracy and missed defects: The new method achieves virtually 100% detection of known defects. The Missing Detection Rate (MDR) has gone from 0.5% with the traditional system to 0% (no known defects escaped detection).

- False alarms: The percentage of false positives has fallen dramatically, from around 5% to around 0.04%. In practice, the old system generated several false alarms, which the human operator then had to check. The new approach reduces false alarms by two orders of magnitude, bringing the maximum number of false defects per board from 10 to 2.
- Inspection speed and throughput: The processing time per card, including image acquisition and inference, has been reduced from approximately 20 seconds to 6 seconds per PCB. This improvement makes the system widely compatible with industrial production rates. Thanks to improved speed and automation, AOI line productivity increases more than three times. In the case tested, the number of boards inspected per hour increased from around 180 using the traditional method to around 600 using the new system.

Model & References	**Dataset**	**Main results**
G-YOLOv8 [128]	DeepPCB dataset and the open PKU PCB defect dataset, both made of color PCB images with labeled bounding boxes for six common defects (missing hole, mouse bite, open, short, spur, spurious copper)	G-YOLOv8 reaches about 125 FPS and slightly improves MAP over the baseline YOLOv8 while using fewer FLOPs, showing that real-time detection of tiny PCB defects is possible without sacrificing accuracy
LPCB-YOLO [131]	Open PKU PCB and DeepPCB datasets: PCB images/image pairs with six common defect types (missing hole, mouse bite, open, short, spur, extra copper), all with bounding-box annotations	On PKU PCB it reaches around 97% precision, 98% recall and 98.4% mAP with fewer parameters and GFLOPs than YOLOv8s; on DeepPCB it keeps accuracy close to YOLOv8 while being smaller and faster, offering a good real-time trade-off for tiny PCB defects
Deep Siamese semantic segmentation network [133]	Public PCB welding defect dataset: 340 registered color image pairs (1280×1024) of OK vs NG boards from an industrial line, with 5 annotated welding defect types (missing solder/component, short, overturned, solder ball)	Outperforms classic and modern segmentation models on weld defects, detecting tiny flaws and staying robust with limited data and brightness/noise changes
Lightweight U-Net-like segmentation plus rule-based solder defect logic [135]	Ad-hoc industrial RGB PCBA dataset: 980 images (2448×2048) with five solder-related defect types	Deployed on a low-power ARM-based edge AI chip, cuts inference from 20s to 6s per image and power from 200 W to 14 W, while sharply reducing missed defects and false alarms and raising line throughput to 600 PCBAs/h

Table 4.1: Summary of image classification and semantic segmentation models for PCB defect detection.

4.4. Reconstruction and anomaly detection for PCB inspection

Anomaly detection refers to the set of techniques employed to find data or events that behave or appear differently from the majority of data, which means to find exceptions or, indeed, anomalies [136]. An autoencoder is a type of neural network used to learn a compact representation of data, that is, to compress information, with an encoder, and then reconstruct it with a decoder. The goal during training is to minimize the difference between input and output so, in other terms, to have a reconstruction as faithful as possible [137]. Autoencoders can be used to perform anomaly detection by being trained on correct data without anomalies. When the network is fed with new data, it attempts to reconstruct it. If the new data lacks anomalies, the reconstruction error will be low. If, on the other hand, the data is significantly different from the data without anomalies the network was trained on, the autoencoder will make a relatively high reconstruction error. Autoencoders are particularly flexible models because their basic goal, reconstructing an input, lends itself to different training schemes. In their most classic form, they are trained in an unsupervised manner: the network is fed with an image as input and is asked to reconstruct the same image without the need for external labels. In anomaly detection applications, for example on PCBs, it is common to train the autoencoder only on defect-free images and, during testing, parts that it cannot reconstruct are interpreted as defects. But autoencoders can also be trained in a supervised/self-supervised manner for denoising, providing a degraded version of the image with noise or synthetic defects as input and using the corresponding clean version as the target. In this case, the model explicitly learns to "clean" the input, and the loss is defined between the output and the uncorrupted reference image.

A significant contribution concerning the employment of autoencoders has been made in 2023 by De Oliveira et al. [138] that studied the problem of tampered assembled PCBs (addition/removal of components) in uncontrolled scenarios, for example illegally modified gas pump cards for fraud. In such cases, photographs may have varying perspectives and lighting, making direct comparison with a reference image difficult. The authors propose tackling the task as one-class anomaly detection, training a deep convolutional autoencoder only on images of the original intact card, without the need to collect examples of all possible fraudulent modifications, which represents one of the key advantages of the autoencoder approach.

From an application perspective, this work highlights how autoencoders can be adapted to challenging industrial and on-site scenarios. The system is flexible enough to operate in

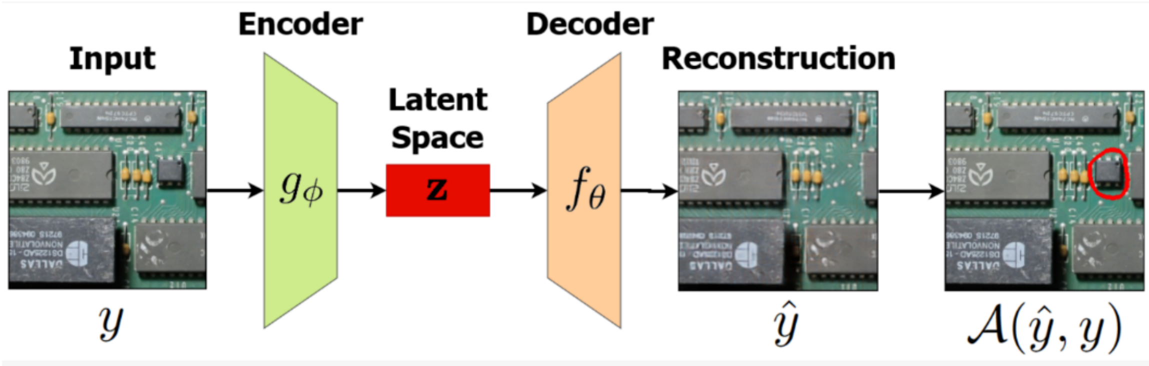


Figure 4.11: Autoencoder employment example for anomaly detection [138].

the field without a fixed camera or specialized lighting, and could assist human inspectors in detecting suspicious changes where manual visual comparison would be slow and prone to error.

Another relevant approach that employs autoencoders, and in particular variational autoencoders, in an unsupervised manner is the one proposed by Ulger et al. [139]. Using an ad-hoc dataset, the authors propose a β-VAE system for inspecting solder joints on assembled PCBs that overcomes many traditional constraints, highlighting the advantage of this approach. Indeed, it requires no special lighting, no golden reference images, and no large collections of labeled defects. The model is trained only on correct solder joints under uncontrolled acquisition conditions. During detection, any abnormal joint will produce a latent code that does not fall within the expected distribution, signaling the presence of a defect without first needing to see that type of defect. Experiments have shown that β-VAE detects defects with high accuracy despite being trained only on normal samples, and does so without requiring specific hardware or feature engineering for acquisition. This is a remarkable result because it means, for example, that in a real-world setting, the system can be installed on an existing line and still detect defective welds, overcoming the limitations of manual visual inspection or methods that require expensive custom equipment.

An example of an enchanced autoencoder trained in a supervised manner instead is the one proposed in 2021 by Kim et al. [140] that proposed a convolutional autoencoder with skip connections, similar to those of U-Net, to avoid the loss of fine details typical of deep autoencoders and improve the inspection of defects in PCBs. The model is trained on pairs of defective patches and golden patches, learning to reconstruct the perfect version of the card and highlighting the defect in the difference image. Thanks to skip connections, accuracy increases from 95% to 98%, with TPR rising from 91% to over 97% while maintaining high specificity and precision, meaning more defects are found without increasing false alarms. In addition, the architecture is lightweight (26.8 BFLOPs per image) and faster than YOLOv4, making it suitable for real-time use on production lines.

In addition to traditional autoencoders, another possibility to perform anomaly detection on PCBs concerns the use of generative adversarial networks (GAN). In 2021 Bougaham et al. [141] proposed a GAN-based system for whole images of automotive PCBA, in a real industrial context where the traditional referential methods with a golden image produced many false alarms due to small tolerable visual variations (reflections, slight printing differences, etc.). The GAN is trained in one-class mode, using only about 360 images of defect-free boards, so that the generator learns to reproduce only the "normal case" without the need for defective samples or detailed labels. In practical terms, the approach translates into a reduction of over 70% in manual inspection time per card (from 8s to approximately 2.24 s), even when working with a highly unbalanced dataset and very few defective images (18 anomalies out of 68 test images). The work therefore shows how a one-class GAN approach, integrated as an AOI post-filter module, can increase the robustness of quality control: it filters out many false alarms due to harmless variability, reports only truly suspicious anomalies to operators, and requires neither special hardware nor a massive campaign of defect collection and annotation.

The discussed works, summarized in Table 4.2, clearly show the relevance of this DL architecture thanks to the academic and industrial advantages they offer. Indeed, not only they provide an increase in performance compared to traditional techniques, but also allow the possibility to perform defect-detection without the actual need of images with defects.

Model	Dataset	Main results	References
Autoencoder	MPI-PCB (public): 1742 high-resolution color images (4096×2816) of one unmodified gas-pump PCB under varying illumination/perspective + 55 modified images with pixel-wise mask	Achieves state of the art anomaly segmentation on assembled PCBs, with accurate localization of tampered components and fewer false positives than existing one-class methods. No need to collect examples of all possible fraudulent modifications	[138]
Variational autoencoder	Ad-hoc private dataset of color solder joints (IC and non-IC): 2513 normal training samples, 567 normal validation samples, 150 test samples (66 abnormal, 84 normal); joints cropped and resized to 64×64	Detects solder joint anomalies with high F1 (0.81) using only normal samples; no special lighting, handcrafted features or golden reference board required, and a single model handles both IC and non-IC joints	[139]
Skip-connected convolutional autoencoder	Public Huang & Wei PCB defect dataset: 693 high-resolution PCB images (10 boards, 6 synthetic defect types);	98% accuracy (TPR >97%) without increasing false alarms, in a lightweight 26.8 BFLOPs architecture faster than YOLOv4, suitable for real-time inline inspection	[140]
Wasserstein GAN	Proprietary, 428 (410 normal and 18 abnormal) high-resolution grayscale images (4500×4340) images	Reduced manual inspection time by > 70% (8s → 2.24 s per board) on a highly unbalanced dataset, by using a one-class GAN as an AOI post-filter that suppresses many false alarms. Requires no special hardware or large defect datasets	[141]

Table 4.2: Summary of autoencoder models for PCB defect detection.

4.5. Advantages and limits of deep learning techniques

The adoption of deep learning techniques in automatic PCB inspection has marked a clear advance over traditional AOI pipelines and methods based on hand-crafted features. A key benefit is that through deep learning it is possible to learn discriminative representations directly from image data, reducing the need for manual feature design and rigid rule tuning (e.g., thresholds or golden-template comparisons). In practice, this end-to-end learning tends to make models more robust to the variability typical of real production, such as changes in lighting, background texture, camera settings, and the diverse visual appearance of defects, where classical approaches often degrade.

Deep learning also enables multiple degrees of inspection within a unified, data-driven framework: it can be configured for fast image-level OK/NG classification, for object detection with bounding boxes to localize and categorize multiple defects on the same board, or for pixel-level semantic segmentation to precisely delineate defect regions and support quantitative measurements. Compared to traditional AOI, which typically relies on task-specific rules (e.g., thresholds, template matching) and often requires re-tuning when layouts or imaging conditions change, deep models can generalize better across variability and scale more naturally to multi-class, irregular defect patterns. However, these advantages come with well-known constraints: training usually requires large, high-quality annotated datasets (often difficult because many defects are rare and imbalanced), significant computational resources, and careful validation to ensure generalization when layouts, cameras, or illumination change. Moreover, deep models can be hard to interpret, behaving like black boxes unless complemented by explainability tools and systematic monitoring. Despite these limitations, deep learning is now widely considered the state of the art for PCB defect inspection and provides a strong basis for future developments such as semi/self-supervised learning to reduce labeling effort, synthetic data to enrich rare defect classes, and lightweight architectures for reliable deployment on edge devices. [131, 142].

5 Conclusions

The main objective of this thesis was to compare traditional AOI approaches with those based on artificial intelligence, both machine learning and deep learning, for PCB defect detection. This work highlighted differences, summarized in Table 5.1, between the different approaches. Traditional AOI systems are based on deterministic algorithms (rule based and template matching) with the main downsides of having low adaptability, especially in presence of different lighting and orientation conditions, and a high false alarm ratio. ML based approaches introduce trainable algorithms based on feature designed by experts. This improves the adaptability and flexibility with respect to fixed rules but efficiency and effectiveness still depend on the quality of the extracted features and the available training data. The current state of the art DL based approaches overcome many of these limitations. Features are no longer defined manually but they are automatically learned by the network, which is also more robust to changing in lighting, orientations and others with respect to the previous techniques. In addition, deep learning has expanded the range of tasks that can be tackled: not only whole-image classification, but also object detection and semantic segmentation of defective regions, allowing for more accurate and detailed identification of anomalies compared to classic AOI. However, it is not always the best choice in every industrial context: in situations with limited computational resources or small training datasets, simpler solutions may be preferable. Traditional or lightweight ML-based methods continue to play an important role when they can meet inspection requirements with less complexity, or when factors such as result interpretability or ease of implementation in the field are a priority.

The latest trends in PCB defect detection research include innovative approaches such as self-supervised learning and multi-modal learning. Self-supervised learning is a learning method that does not require human labels: the model learns to recognize patterns and anomalies from unlabeled data. This approach is particularly useful in the PCB context, where the availability of labeled images is often limited, especially for rare defects. Multi-modal learning combines different types of information (such as images, audio signals and text) to improve the model's semantic understanding. Applied to PCB inspection, for example, it allows images to be enriched with textual descriptions of defects, facilitating

Characteristic	**Traditional AOI**	**Machine Learning approaches**	**Deep Learning approaches**
Feature extraction	Manual features/criteria (rules, templates, image processing)	Manual (hand-crafted descriptors)	Automatic (end-to-end representation learning)
Robustness to noise / variations	Low–medium (depends on controlled conditions, calibration, and alignment)	Medium (depends on the descriptor and pre-processing)	Medium–high (if training covers variability; sensitive to domain shift)
Adaptability	Limited (re-tuning rules/templates for PCB and setup variations)	Limited (often requires feature redesign and re-training)	High
False-positive handling	Often critical (tolerable variations may be flagged as defects)	Improved but still present	Often reduced
Types of defects	Known, well-modelled defects; fragile under variability	Defects with relatively predictable patterns (tied to chosen features)	Complex and variable defects; better generalization to variants

Table 5.1: Comparative summary of classical AOI, traditional ML, and deep learning approaches for PCB defect detection.

learning with few examples. Moreover, edge-oriented and edge–cloud architectures are emerging. In these scenarios, lighter models are run directly on local devices to ensure real-time inference on the line, as previously seen describing the approach of Li et al. [135], which aims to deploy a semantic segmentation pipeline on an Edge Intelligent System through optimizations such as quantization and efficient convolutions, drastically reducing inspection times and false alarms in production. In more comprehensive edge-cloud systems, on the other hand, the cloud can complement the edge by handling storage, historical analysis, and model retraining/updating. Overall, this direction enables rapid and efficient inspections and facilitates progressive adaptation to new defects and process variations [143].

Bibliography

[1] Heiner Lasi, Peter Fettke, Hans-Georg Kemper, Thomas Feld, and Michael Hoffmann. Industry 4.0. *Business & Information Systems Engineering*, 6(4):239–242, August 2014.

[2] Francisco Folgado, David Calderón, Isaías González, and Antonio Calderón. Review of Industry 4.0 from the Perspective of Automation and Supervision Systems: Definitions, Architectures and Recent Trends. *Electronics*, 13(4):782, February 2024.

[3] Shun-Feng Su, Imre J. Rudas, Jacek M. Zurada, Meng Joo Er, Jyh-Horng Chou, and Daeil Kwon. Industry 4.0: A Special Section in IEEE Access. *IEEE Access*, 5:12257–12261, 2017.

[4] Mario Hermann, Tobias Pentek, and Boris Otto. Design Principles for Industrie 4.0 Scenarios: A Literature Review, 2015.

[5] Plattform Industrie 4.0 - What is Industrie 4.0?

[6] Hu-Chen Liu, Ran Liu, Xiuzhu Gu, and Miying Yang. From total quality management to Quality 4.0: A systematic literature review and future research agenda. *Frontiers of Engineering Management*, 10(2):191–205, June 2023.

[7] Sarvesh Sundaram and Abe Zeid. Artificial Intelligence-Based Smart Quality Inspection for Manufacturing. *Micromachines*, 14(3):570, February 2023.

[8] Snehasis Sahoo and Cheng-Yao Lo. Smart manufacturing powered by recent technological advancements: A review. *Journal of Manufacturing Systems*, 64:236–250, July 2022.

[9] Nithya Sambasivan, Shivani Kapania, Hannah Highfill, Diana Akrong, Praveen Paritosh, and Lora M Aroyo. “Everyone wants to do the model work, not the data work”: Data Cascades in High-Stakes AI. In *Proceedings of the 2021 CHI conference on human factors in computing systems*, Chi ’21, Yokohama, Japan, 2021. Association for Computing Machinery.

[10] Yasuyo Kita, Hiroshi Ishikawa, and Takeshi Masuda. Guest Editorial: Machine

Vision Applications. *International Journal of Computer Vision*, 122(2):191–192, April 2017.

[11] W.S. Lee, V. Alchanatis, C. Yang, M. Hirafuji, D. Moshou, and C. Li. Sensing technologies for precision specialty crop production. *Computers and Electronics in Agriculture*, 74(1):2–33, October 2010.

[12] Maria Tzampazaki, Charalampos Zografos, Eleni Vrochidou, and George A. Papakostas. Machine Vision—Moving from Industry 4.0 to Industry 5.0. *Applied Sciences*, 14(4):1471, February 2024.

[13] Zhonghe Ren, Fengzhou Fang, Ning Yan, and You Wu. State of the Art in Defect Detection Based on Machine Vision. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 9(2):661–691, March 2022.

[14] Qinbang Zhou, Renwen Chen, Bin Huang, Chuan Liu, Jie Yu, and Xiaoqing Yu. An Automatic Surface Defect Inspection System for Automobiles Using Machine Vision Methods. *Sensors*, 19(3):644, February 2019.

[15] Rui Carrilho, Ehsan Yaghoubi, José Lindo, Kailash Hambarde, and Hugo Proença. Toward Automated Fabric Defect Detection: A Survey of Recent Computer Vision Approaches. *Electronics*, 13(18):3728, September 2024.

[16] Dorián László Galata, Lilla Alexandra Mészáros, Nikolett Kállai-Szabó, Edina Szabó, Hajnalka Pataki, György Marosi, and Zsombor Kristóf Nagy. Applications of machine vision in pharmaceutical technology: A review. *European Journal of Pharmaceutical Sciences*, 159:105717, April 2021.

[17] Alexander Schoo, Robin Moschner, Jens Hülsmann, and Arno Kwade. Coating Defects of Lithium-Ion Battery Electrodes and Their Inline Detection and Tracking. *Batteries*, 9(2):111, February 2023.

[18] Katleho Masita, Ali Hasan, Thokozani Shongwe, and Hasan Abu Hilal. Deep learning in defects detection of PV modules: A review. *Solar Energy Advances*, 5:100090, 2025.

[19] Yi-Hung Liu, Yan-Chen Liu, and Yen-Zen Chen. High-speed inline defect detection for TFT-LCD array process using a novel support vector data description. *Expert Systems with Applications*, 38(5):6222–6231, May 2011.

[20] Daiki Tomita and Yue Bao. Inspection of Defective Glass Bottle Mouths Using Machine Learning. *Journal of Imaging*, 11(4):105, March 2025.

[21] Roland T. Chin and Charles A. Harlow. Automated Visual Inspection: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-4(6):557–573, November 1982.

[22] Ian Grout. PCB Design. In *Digital Systems Design with FPGAs and CPLDs*, pages 123–176. Elsevier, 2008.

[23] Khushi Singh, Shubhangi Kharche, Aakash Chauhan, and Pranali Salvi. PCB Defect Detection Methods: A Review of Existing Methods and Potential Enhancements. *Journal of Engineering Science and Technology Review*, 17(1):156–167, February 2024.

[24] M. Moganti and F. Ercal. Automatic PCB inspection systems. *IEEE Potentials*, 14(3):6–10, September 1995.

[25] TIMOTHY S. NEWMAN and ANIL K. JAIN. A Survey of Automated Visual Inspection. June 1994.

[26] Kerem Aras, Syed Saad Saif, Alessandro Giuseppi, and Vedat Coskun. Automated Optical Inspection for Quality Control in PCBA assembly lines: a case study for Point of Sale Devices Production Lines. In *2024 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pages 1–8, Istanbul, Turkiye, May 2024. IEEE.

[27] O. Hecht and G. Dishon. Automatic optical inspection (AOI) of thin film multichip modules. In *8th IEEE/CHMT International Conference on Electronic Manufacturing Technology Symposium*, pages 130–136, Baveno, Italy, 1990. IEEE.

[28] Ankit Bharatbhai Goti. Automated Optical Inspection (AOI) Based on IPC Standards.

[29] Shruti Rajiv Vaidya. *AUTOMATIC OPTICAL INSPECTION-BASED PCB FAULT DETECTION USING IMAGE PROCESSING.* Master of Science in Electrical and Computer Engineering, Michigan Technological University, Houghton, Michigan, 2023.

[30] Abd Al Rahman M. Abu Ebayyeh and Alireza Mousavi. A Review and Analysis of Automatic Optical Inspection and Quality Monitoring Methods in Electronics Industry. *IEEE Access*, 8:183192–183271, 2020.

[31] Chuan-Hao Liu, Wei-Lun Lin, and Fan-Shuo Tseng. Intelligent detection methods for miniature defects in metallic materials. *Array*, 27:100430, 2025.

[32] Fupei Wu and Xianmin Zhang. An inspection and classification method for chip solder joints using color grads and Boolean rules. *Robotics and Computer-Integrated Manufacturing*, 30(5):517–526, 2014.

[33] Richard Mc. TRI White Paper – Introduction to AOI Technology.

[34] National Instruments. IMAQ Vision Concepts Manual. volume 322916A-01, Austin, TX, USA, 2000.

[35] Y. V. Chavan, D. K. Mishra, D. S. Bormane, A. D. Shaligram, and N. S. Mujumdar. 2-D and 3-D lens model for computer vision. *Journal of Optics*, 46(2):108–115, June 2017.

[36] Chauhan, Vedang, Fernando, Heshan, and Surgenor, Brian. Effect of Illumination Techniques on Machine Vision Inspection for Automated Assembly Machines. 2014.

[37] Yongbing Zhou, Minghao Yuan, Jian Zhang, Guofu Ding, and Shengfeng Qin. Review of vision-based defect detection research and its perspectives for printed circuit board. *Journal of Manufacturing Systems*, 70:557–578, October 2023.

[38] Eid M Taha, E Emary, and Khalid Moustafa. Automatic Optical Inspection for PCB Manufacturing: a Survey. 2014.

[39] Madhav Moganti, Fikret Ercal, Cihan H. Dagli, and Shou Tsunekawa. Automatic PCB Inspection Algorithms: A Survey. *Computer Vision and Image Understanding*, 63(2):287–313, March 1996.

[40] Zihan He, Yudong Lian, Yulei Wang, and Zhiwei Lu. A Comprehensive Review of Research on Surface Defect Detection of PCBs Based on Machine Vision. *Results in Engineering*, page 106437, July 2025.

[41] Hongwei Xie, Yongcong Kuang, and Xianmin Zhang. A high speed AOI algorithm for chip component based on image difference. 2009 IEEE International Conference on Information and Automation, ICIA 2009, pages 969 – 974, July 2009.

[42] A.M. Darwish and A.K. Jain. A rule based approach for visual pattern inspection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(1):56–68, January 1988.

[43] Sun, Yung-Nien and Tsai, Ching-Tsorng. A new model-based approach for industrial visual inspection. *Pattern Recognit.*, pages 1327–1336, 1992.

[44] Q.-Z. Ye and P.E. Danielsson. Inspection of Printed Circuit Boards by Connec-

tivity Preserving Shrinking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):737–742, September 1988.

[45] Robert Thibadeau. Printed Circuit Board Inspection.

[46] Jon R. Mandeville. Novel method for analysis of printed circuit images. *IBM Journal of Research and Development*, 29(1):73–86, 1985.

[47] S.H Indera Putera and Z. Ibrahim. Printed circuit board defect detection using mathematical morphology and MATLAB image processing tools. In *2010 2nd International Conference on Education Technology and Computer*, pages V5–359–V5–363, Shanghai, China, June 2010. IEEE.

[48] Luis Augusto Libório Oliveira Fonseca, Yuzo Iano, Gabriel Gomes De Oliveira, Gabriel Caumo Vaz, Giulliano Paes Carnielli, Júlio César Pereira, and Rangel Arthur. Automatic printed circuit board inspection: a comprehensible survey. *Discover Artificial Intelligence*, 4(1):10, February 2024.

[49] Madhav Moganti, Fikret Ercal, Cihan H. Dagli, and Shou Tsunekawa. Automatic PCB Inspection Algorithms: A Survey. *Computer Vision and Image Understanding*, 63(2):287–313, March 1996.

[50] D. B. Anitha and Mahesh Rao. A survey on defect detection in bare PCB and assembled PCB using image processing techniques. In *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pages 39–43, Chennai, March 2017. IEEE.

[51] Beiwen Li, Yatong An, David Cappelleri, Jing Xu, and Song Zhang. High-accuracy, high-speed 3D structured light imaging techniques and potential applications to intelligent robotics. *International Journal of Intelligent Robotics and Applications*, 1(1):86–103, February 2017.

[52] U.R. Dhond and J.K. Aggarwal. Structure from stereo-a review. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1489–1510, December 1989.

[53] Carlos Madrigal, John Branch, Alejandro Restrepo, and Domingo Mery. A Method for Automatic Surface Inspection Using a Model-Based 3D Descriptor. *Sensors*, 17(10):2262, October 2017.

[54] Joaquim Salvi, Sergio Fernandez, Tomislav Pribanic, and Xavier Llado. A state of the art in structured light patterns for surface profilometry. *Pattern Recognition*, 43(8):2666–2680, August 2010.

[55] Song Zhang. Recent progresses on real-time 3D shape measurement using digital fringe projection techniques. *Optics and Lasers in Engineering*, 48(2):149–158, February 2010.

[56] Matthias Breier, Philipp Moller, Wei Li, Marcel Bosling, Thomas Pretz, and Dorit Merhof. Accurate laser triangulation using a perpendicular camera setup to assess the height profile of PCBs. In *2015 IEEE International Conference on Industrial Technology (ICIT)*, pages 1613–1618, Seville, March 2015. IEEE.

[57] Pedro M A Vitoriano and Tito G Amaral. SMT Magazine, June 2016.

[58] Automated Optical Inspection in PCB Manufacturing: How AOI Systems Improve Quality and Efficiency - TSTRONIC, October 2025.

[59] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[60] I-Chun Chen, Rey-Chue Hwang, and Huang-Chu Huang. PCB Defect Detection Based on Deep Learning Algorithm. *Processes*, 11(3):775, March 2023.

[61] Chien-Yi Huang, I-Chen Lin, and Yuan-Lien Liu. Applying Deep Learning to Construct a Defect Detection System for Ceramic Substrates. *Applied Sciences*, 12(5):2269, February 2022.

[62] Rong-Qing Qiu, Mu-Lin Tsai, Yu-Wen Chen, Shivendra Pratap Singh, and Cheng-Yao Lo. Integrated Automatic Optical Inspection and Image Processing Procedure for Smart Sensing in Production Lines. *Sensors*, 24(5):1619, March 2024.

[63] Shuixin Deng, Lei Deng, Ting Sun, Shijie Yu, Li Wang, Baohua Chen, Hao Hu, Yusen Xie, Hanxi Yin, Junwei Xiao, Xinglong Cui, Yeyu Fu, Xuewei Tang, Ruirui Song, Lin Li, Shanpeng Xiao, Yuan Li, and Yizheng Li. EEDD: Edge-Guided Energy-Based PCB Defect Detection. *Electronics*, 12(10):2306, May 2023.

[64] Qianru Zhang, Meng Zhang, Chinthaka Gamanayake, Chau Yuen, Zehao Geng, Hirunima Jayasekara, Chia-wei Woo, Jenny Low, Xiang Liu, and Yong Liang Guan. Deep learning based solder joint defect detection on industrial printed circuit board X-ray images. *Complex & Intelligent Systems*, 8(2):1525–1537, April 2022.

[65] Chien-Yi Huang, I-Chen Lin, and Yuan-Lien Liu. Applying Deep Learning to Construct a Defect Detection System for Ceramic Substrates. *Applied Sciences*, 12(5):2269, February 2022.

[66] Alberto Borboni, Karna Vishnu Vardhana Reddy, Irraivan Elamvazuthi, Maged S.

AL-Quraishi, Elango Natarajan, and Syed Saad Azhar Ali. The Expanding Role of Artificial Intelligence in Collaborative Robots for Industrial Applications: A Systematic Review of Recent Works. *Machines*, 11(1):111, January 2023.

[67] Soufiane Douimia, Abdelghani Bekrar, Abdessamad Ait El Cadi, Yassin El Hillali, and David Fillon. Machine learning and deep learning applications in the automotive manufacturing industry: A systematic literature review and industry insights. *Robotics and Computer-Integrated Manufacturing*, 96:103034, December 2025.

[68] Yongkun Deng, Chenghao Zhang, Nan Yang, and Huaming Chen. FocalMatch: Mitigating Class Imbalance of Pseudo Labels in Semi-Supervised Learning. *Applied Sciences*, 12(20):10623, October 2022.

[69] Raffaele Pugliese, Stefano Regondi, and Riccardo Marini. Machine learning-based approach: global trends, research directions, and regulatory standpoints. *Data Science and Management*, 4:19–29, December 2021.

[70] İbrahim Yazici, Ibraheem Shayea, and Jafri Din. A survey of applications of artificial intelligence and machine learning in future mobile networks-enabled systems. *Engineering Science and Technology, an International Journal*, 44:101455, August 2023.

[71] Haofu Liao, S. Kevin Zhou, and Jiebo Luo. Chapter 3 - Classification: lesion and disease recognition. In Haofu Liao, S. Kevin Zhou, and Jiebo Luo, editors, *Deep network design for medical image computing*, The MICCAI society book series, pages 27–58. Academic Press, 2023.

[72] Masoud Aliramezani, Charles Robert Koch, and Mahdi Shahbakhti. Modeling, diagnostics, optimization, and control of internal combustion engines via modern machine learning techniques: A review and future directions. *Progress in Energy and Combustion Science*, 88:100967, January 2022.

[73] Thorsten Wuest, Daniel Weimer, Christopher Irgens, and Klaus-Dieter Thoben. Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*, 4(1):23–45, January 2016.

[74] Huawei Technologies Co., Ltd. *Artificial Intelligence Technology*. Springer Nature Singapore, Singapore, 2023.

[75] Confusion Matrix - an overview | ScienceDirect Topics.

[76] Yongbing Zhou, Minghao Yuan, Jian Zhang, Guofu Ding, and Shengfeng Qin. Re-

view of vision-based defect detection research and its perspectives for printed circuit board. *Journal of Manufacturing Systems*, 70:557–578, October 2023.

[77] A Beginner's Guide to Feature Engineering in Machine Vision, June 2025.

[78] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893, San Diego, CA, USA, 2005. IEEE.

[79] Histogram of oriented gradients, March 2025.

[80] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, July 2002.

[81] Local binary patterns, September 2025.

[82] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face Recognition with Local Binary Patterns. In Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Dough Tygar, Moshe Y. Vardi, Gerhard Weikum, Tomás Pajdla, and Jiří Matas, editors, *Computer Vision - ECCV 2004*, volume 3021, pages 469–481. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[83] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995.

[84] What Is Support Vector Machine? | IBM, December 2023.

[85] Raquel Rodríguez-Pérez and Jürgen Bajorath. Evolution of Support Vector Machine and Regression Modeling in Chemoinformatics and Drug Discovery. *Journal of Computer-Aided Molecular Design*, 36:1–8, May 2022.

[86] Hiroaki Hagi, Yuji Iwahori, Shinji Fukui, Yoshinori Adachi, and M.K. Bhuyan. Defect Classification of Electronic Circuit Board Using SVM based on Random Sampling. *Procedia Computer Science*, 35:1210–1218, 2014.

[87] Chia-Te Liao, Wen-Hao Lee, and Shang-Hong Lai. A Flexible PCB Inspection System Based on Statistical Learning. *Journal of Signal Processing Systems*, 67(3):279–290, June 2012.

[88] Zhongqiu Zhang, Xiaodong Wang, Shan Liu, Li Sun, Liye Chen, and Yangming Guo. An Automatic Recognition Method for PCB Visual Defects. In *2018 International*

Conference on Sensing,Diagnostics, Prognostics, and Control (SDPC), pages 138–142, Xi'an, China, August 2018. IEEE.

[89] Hao Wu, Xianmin Zhang, Hongwei Xie, Yongcong Kuang, and Gaofei Ouyang. Classification of Solder Joint Using Feature Selection Based on Bayes and Support Vector Machine. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 3(3):516–522, March 2013.

[90] Ivana Cavar Semanjski. Chapter 5 - Data analytics. In Ivana Cavar Semanjski, editor, *Smart Urban Mobility*, pages 121–170. Elsevier, January 2023.

[91] Yasha Hasija. Chapter 9 - A machine learning approach to bioinformatics. In Yasha Hasija, editor, *All About Bioinformatics*, pages 203–224. Academic Press, January 2023.

[92] B.C. Jiang, C.C. Wang, and P.L. Chen. Logistic regression tree applied to classify PCB golden finger defects. *The International Journal of Advanced Manufacturing Technology*, 24(7-8):496–502, October 2004.

[93] Xie Hongwei, Zhang Xianmin, Kuang Yongcong, and Ouyang Gaofei. Solder Joint Inspection Method for Chip Component Using Improved AdaBoost and Decision Tree. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 1(12):2018–2027, December 2011.

[94] Random Forest - an overview | ScienceDirect Topics.

[95] Guo-Feng Fan, Liu-Zhen Zhang, Meng Yu, Wei-Chiang Hong, and Song-Qiao Dong. Applications of random forest in multivariable response surface for short-term load forecasting. *International Journal of Electrical Power & Energy Systems*, 139:108073, July 2022.

[96] Eun Hye Yuk, Seung Hwan Park, Cheong-Sool Park, and Jun-Geol Baek. Feature-Learning-Based Printed Circuit Board Inspection via Speeded-Up Robust Features and Random Forest. *Applied Sciences*, 8(6):932, June 2018.

[97] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. Statistical learning. In *An introduction to statistical learning: with applications in R*, pages 15–57. Springer New York, New York, NY, 2013.

[98] What is the k-nearest neighbors algorithm? | IBM, October 2021.

[99] Meiju Liu, Lingyan Li, and Wenbo Guo. The Improved K-nearest Neighbor Solder Joints Defect Detection.

[100] Haibo He and Edwardo A. Garcia. Learning from Imbalanced Data. *IEEE Trans. on Knowl. and Data Eng.*, 21(9):1263–1284, 2009.

[101] K-Nearest Neighbor Classifier - an overview | ScienceDirect Topics.

[102] Hajar Hakkoum, Ali Idri, and Ibtissam Abnane. Global and local interpretability techniques of supervised machine learning black box models for numerical medical data. *Engineering Applications of Artificial Intelligence*, 131:107829, May 2024.

[103] Stable and actionable explanations of black-box models through factual and counterfactual rules | Data Mining and Knowledge Discovery.

[104] Overview of Deep Learning. In Ltd. Huawei Technologies Co., editor, *Artificial Intelligence Technology*, pages 87–122. Springer Nature, Singapore, 2023.

[105] Putra Sumari Laith Abualigah, Saqib Jamal Syed. A Novel Deep Learning Pipeline Architecture based on CNN to Detect Covid-19 in Chest X-ray Images. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12(6):2001–2011, April 2021.

[106] Christian Janiesch, Patrick Zschech, and Kai Heinrich. Machine learning and deep learning. *Electronic Markets*, 31(3):685–695, September 2021.

[107] Shengping Lv, Bin Ouyang, Zhihua Deng, Tairan Liang, Shixin Jiang, Kaibin Zhang, Jianyu Chen, and Zhuohui Li. A dataset for deep learning based detection of printed circuit board surface defect. *Scientific Data*, 11(1):811, July 2024.

[108] Gianmauro Fontana, Maurizio Calabrese, Leonardo Agnusdei, Gabriele Papadia, and Antonio Del Prete. SolDef_ai: An Open Source PCB Dataset for Mask R-CNN Defect Detection in Soldering Processes of Electronic Components. *Journal of Manufacturing and Materials Processing*, 8(3):117, May 2024.

[109] tangsanli5201. tangsanli5201/DeepPCB, October 2025.

[110] Bing Hu and Jianhui Wang. Detection of PCB Surface Defects With Improved Faster-RCNN and Feature Pyramid Network. *IEEE Access*, 8:108335–108345, 2020.

[111] Xinting Liao, Shengping Lv, Denghui Li, Yong Luo, Zichun Zhu, and Cheng Jiang. YOLOv4-MN3 for PCB Surface Defect Detection. *Applied Sciences*, 11(24):11701, January 2021.

[112] Tryambak Gangopadhyay, Sungmin Hong, Sujoy Roy, Yash Shah, and Lin Lee Cheong. Benchmarking framework for anomaly localization: Towards real-world

deployment of automated visual inspection. *Journal of Manufacturing Systems*, 69:64–75, August 2023.

[113] Ahmed Jawad Rashid, Mohammad Aman Ullah, Adiba Isfara, Nadim Ahmed, Md. Mamun Mian, and Md. Mashur Shalehin. PCB-defect: An annotated dataset for surface defect detection in printed circuit boards. *Data in Brief*, 64:112296, February 2026.

[114] Danqing Kang. youtang1993/MeiweiPCB, December 2024.

[115] Raffaele Mineo, Amelia Sorrenti, Robin Faro, Gabriele Mineo, Francesco Cancelliere, and Alberto Faro. PCB-SAID: A Low-Cost Camera-Based Dataset for Few-Shot SMD Assembly Inspection.

[116] Si-Tung Chung, Wen-Jyi Hwang, and Tsung-Ming Tai. Keypoint-Based Automated Component Placement Inspection for Printed Circuit Boards. *Applied Sciences*, 13(17):9863, August 2023.

[117] Feature Extraction Network - an overview | ScienceDirect Topics.

[118] Moez Krichen. Convolutional Neural Networks: A Survey. *Computers*, 12(8):151, August 2023.

[119] Veysel Kocaman, Ofer Shir, and Thomas Bäck. *Improving Model Accuracy for Imbalanced Image Classification Tasks by Adding a Final Batch Normalization Layer: An Empirical Study.* November 2020.

[120] Luca Martiri, Andrea Moschetti, Emilia Lenzi, Marco Zanoni, Loredana Cristaldi, Letizia Tanca, and Davide Martinenghi. A Data Pipeline to Classify PCB Welding Defects on Noisy Data. In *2025 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pages 1–6, Chemnitz, Germany, May 2025. IEEE.

[121] Zhihao Su, Afzan Adam, Mohammad Faidzul Nasrudin, and Anton Satria Prabuwono. Proposal-Free Fully Convolutional Network: Object Detection Based on a Box Map. *Sensors*, 24(11):3529, May 2024.

[122] Momina Liaqat Ali and Zhou Zhang. The YOLO Framework: A Comprehensive Review of Evolution, Applications, and Benchmarks in Object Detection. *Computers*, 13(12):336, December 2024.

[123] Jiayang An and Zhichao Shi. YOLOv8n-Enhanced PCB Defect Detection: A

Lightweight Method Integrating Spatial–Channel Reconstruction and Adaptive Feature Selection. *Applied Sciences*, 14(17):7686, August 2024.

[124] Luca Martiri, Andrea Moschetti, and Loredana Cristaldi. YOLO-Based Object Detection of PCB Components Under Degraded Image Conditions.

[125] Jianan Wang, Xin Xie, Guoying Liu, and Liang Wu. A Lightweight PCB Defect Detection Algorithm Based on Improved YOLOv8-PCB. *Symmetry*, 17(2):309, February 2025.

[126] Yan Zhang, Haifeng Zhang, Qingqing Huang, Yan Han, and Minghang Zhao. DsP-YOLO: An anchor-free network with DsPAN for small object detection of multiscale defects. *Expert Systems with Applications*, 241:122669, November 2023.

[127] Zhongmin Huangfu, Shuqing Li, and Luoheng Yan. Ghost-YOLO v8: An Attention-Guided Enhanced Small Target Detection Algorithm for Floating Litter on Water Surfaces. *Computers, Materials & Continua*, 80(3):3713–3731, 2024.

[128] Tangyu Ji, Qian Zhao, Kang An, Dandan Liu, Wentao Yu, and Shuang Liang. A Real-Time PCB Defect Detection Model Based on Enhanced Semantic Information Fusion, December 2023.

[129] Ligang Wu, Liang Zhang, and Qian Zhou. Printed Circuit Board Quality Detection Method Integrating Lightweight Network and Dual Attention Mechanism. *IEEE Access*, 10:87617–87629, 2022.

[130] Zhe Quan and Jun Sun. A Feature-Enhanced Small Object Detection Algorithm Based on Attention Mechanism. *Sensors*, 25(2):589, January 2025.

[131] Haiyan Zhang, Yazhou Li, Dipu Md Sharid Kayes, Zhaoyu Song, and Yuanyuan Wang. Research on PCB defect detection algorithm based on LPCB-YOLO. *Frontiers in Physics*, 12:1472584, January 2025.

[132] What Is Semantic Segmentation? | IBM, September 2023.

[133] Zhigang Ling, Aoran Zhang, Dexin Ma, Yuxin Shi, and He Wen. Deep Siamese Semantic Segmentation Network for PCB Welding Defect Detection. *IEEE Transactions on Instrumentation and Measurement*, 71:1–11, 2022.

[134] Zhongke Li and Xiaofang Liu. Soldering Defect Segmentation Method for PCB on Improved UNet. *Applied Sciences*, 14(16):7370, August 2024.

[135] Yang Li, Xiao Wang, Zhifan He, Ze Wang, Ke Cheng, Sanchuan Ding, Yijing Fan, Xiaotao Li, Yawen Niu, Shanpeng Xiao, Zhenqi Hao, Bin Gao, and Huaqiang Wu.

Industry-Oriented Detection Method of PCBA Defects Using Semantic Segmentation Models. *IEEE/CAA Journal of Automatica Sinica*, 11(6):1438–1446, June 2024.

[136] Murad A. Rassam, Anazida Zainal, and Mohd Aizaini Maarof. Advancements of Data Anomaly Detection Research in Wireless Sensor Networks: A Survey and Open Issues. *Sensors*, 13(8):10087–10122, August 2013.

[137] What Is an Autoencoder? | IBM, November 2023.

[138] Diulhio Candido De Oliveira, Bogdan Tomoyuki Nassu, and Marco Aurelio Wehrmeister. Image-Based Detection of Modifications in Assembled PCBs with Deep Convolutional Autoencoders. *Sensors*, 23(3):1353, January 2023.

[139] Furkan Ulger, Seniha Esen Yuksel, and Atila Yilmaz. Anomaly Detection for Solder Joints Using -VAE. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 11(12):2214–2221, December 2021.

[140] Jungsuk Kim, Jungbeom Ko, Hojong Choi, and Hyunchul Kim. Printed Circuit Board Defect Detection Using Deep Learning via A Skip-Connected Convolutional Autoencoder. *Sensors*, 21(15):4968, July 2021.

[141] Arnaud Bougaham, Adrien Bibal, Isabelle Linden, and Benoıt Frenay. GanoDIP - GAN Anomaly Detection through Intermediate Patches: a PCBA Manufacturing Case.

[142] Md Rakibul Islam, Shahina Begum, and Mobyen Uddin Ahmed. Interpretable Self-Supervised Learning for Fault Identification in Printed Circuit Board Assembly Testing. *Applied Sciences*, 15(18):10080, September 2025.

[143] Zihan He, Yudong Lian, Yulei Wang, and Zhiwei Lu. A comprehensive review of research on surface defect detection of PCBs based on machine vision. *Results in Engineering*, 27:106437, September 2025.

List of Figures

List of Tables

Acknowledgements

This thesis marks the end of a long and difficult journey that even I didn't think I would be able to finish. The list of people who made all this possible, in one way or another, is long, but I will try to keep it short.

I would like to thank my parents for always being the backbone of my life and for supporting me every day, in every way. It's not in my nature to say it often, and maybe I've never even said it out loud, but I love you, and thank you.

To my sister, I know I often joke about being the guiding light in your life, but the truth is exactly the opposite. Trying to be a good older brother and a good role model, as much as possible, has always given me a purpose, even on days and nights when it was hard to find one.

To my friends Gianluca, Luca, and Matteo, for being by my side for years, some of you practically since birth. Thank you for every night out, every laugh, and every hangover we've shared over the years. For many reasons, it's been a while since we've been there, but the Rolling will always be our bar: the place where we could change the music whenever and however we wanted, and pay ten euros instead of fifty.

To Federica, for giving me a level of support I didn't think a friend could provide. I never say it, and maybe I show it even less, but I am so grateful to have you in my life. I know I can always rely on you, whether we're talking about stupid things or serious ones. I know you will notice, just as you always have, when I stop replying, when I isolate myself, and when I need the kind of help and understanding that only you can give, because only you know certain parts of my mind.

Finally, I would also like to thank Anna. You won't be with me today because life has taken its course, but you have always been there for me since the moment we met, and I have shared the greatest moments of my life with you. No amount of words can even come close to describing the importance you've had in my life. But, even though I always second-guess everything, I am sure I wouldn't have arrived here today without your support. I would have collapsed in on myself a long time ago, so thank you for keeping me standing.