In-situ defect detection of metal Additive Manufacturing: an integrated framework

In-situ defect detection of metal Additive Manufacturing: an integrated framework

Davide Cannizzaro (*Student Member, IEEE*), Antonio Giuseppe Varrella, Stefano Paradiso, Roberta Sampieri, Yukai Chen (*Member, IEEE*), Alberto Macii (*Senior Member, IEEE*), Edoardo Patti (*Member, IEEE*) and Santa Di Cataldo (*Member, IEEE*)

Abstract—Metal Additive Manufacturing (AM) is a pillar of the Industry 4.0, with many attractive advantages compared to traditional subtractive fabrication technologies. However, there are many quality issues that can be an obstacle for mass production. The in-situ camera-based monitoring and detection of defects, taking advantage of the layer-by-layer nature of the build, can be an effective solution to this problem. In this context, the use of Computer Vision and Machine Learning algorithms have a very important role. Nonetheless, they are up to this date limited by the scarcity of data for the training, as well as by the difficulty of accessing and integrating the AM process data throughout the fabrication. To tackle this problem, this paper proposes a system for in-situ monitoring that analyses images from an off-axis camera mounted on top of the machine to detect the arising defects in real-time, with automated generation of synthetic images based on Generative Adversarial Network (GAN) for dataset augmentation purposes. The computing functionalities are embedded into a holistic distributed AM platform allowing the collection, integration and storage of data at all stages of the AM pipeline.

Index Terms—Industry 4.0, Additive Manufacturing, Powder Bed Fusion, Computer Vision, Machine learning

1 Introduction

Additive manufacturing (AM) is the industrial name of 3D printing, referring to a large family of computer-directed technologies that grow an object by joining materials layer upon layer starting from a 3D model. This process has shown sensible advantages in terms of development time, production steps, costs and use of material, as well as much higher flexibility than traditional subtractive technologies. Thanks to these characteristics, it is typically considered one of the pillars of the Industry 4.0 revolution.

The flexibility of AM stems from the possibility of implementing a fully digitalized and tool-free production flow from design to final object, without intermediate steps like the creation of molds or dies. The typical phases of this flow are shown in Fig. 1(a): the process starts from a CAD model, that is a full 3D representation of the characteristics of the object to be manufactured. Then, the CAD is first converted into a stereolythography (STL) file, a simplified triangular mesh representation of CAD geometry, and then into a series of thin layers, with a process that goes by the name of *slicing*. This allows to produce machining instructions tailored to a specific AM system, that fabricates the object in a layer-by-layer fashion. The process ends with an optional subtractive finishing process, to achieve the best surface resolutions.

Depending on the target material and involved energy source, an AM system (phase 4 of Fig. 1(a)) entails different types of processes. One of the most popular in industry is Powder Bed Fusion (PBF), that involves the spreading of powder material on top of the previous layers by means of a roller or recoater, with a reservoir continuously providing fresh material supply during the layering. A heat source, either a laser or an electron beam, selectively melts together each layer of powder using high-temperature [1].

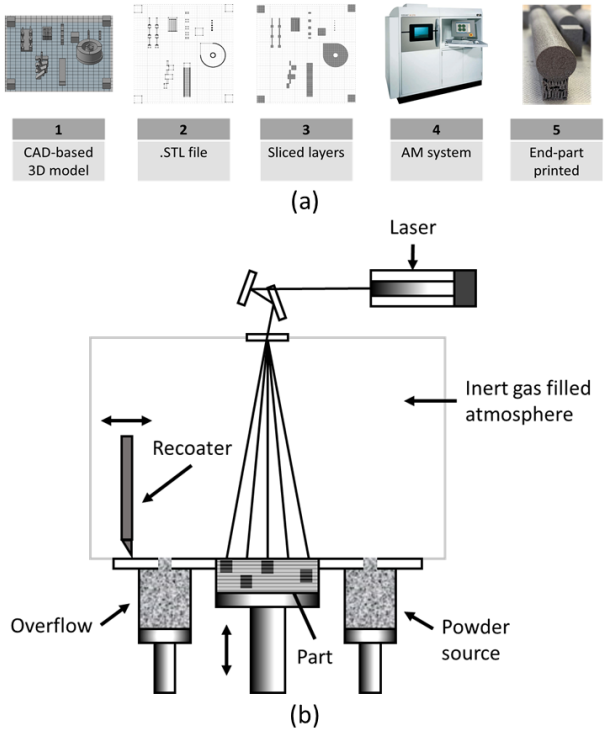


Fig. 1: Additive Manufacturing in industry. (a) Main phases of a typical process. (b) Scheme of laser Powder Bed Fusion.

- *D. Cannizzaro, A. G. Varrella, Y. Chen, A. Macii, E. Patti and S. Di Cataldo are with the Dept. of Control and Computer Engineering Politecnico di Torino, Italy. E-mails: {name.surname}@polito.it*
- *S. Paradiso and R. Sampieri are with Stellantis, Torino, Italy. E-mails: {name.surname}@stellantis.com*

For applications targeting the production of metal components, the most diffused PBF technique is Selective Laser Melting (SLM) or Direct Metal Laser Sintering (DMLS), that exploits a laser as the heat source to either melt or sinter the metal powder together (see a schematic representation in Fig. 1(b)).

Thanks to its many advantages, AM is pushing forward innovative designs and applications in highly competitive fields like aerospace, automotive and medical. In aerospace and automotive, AM is helping suppliers and companies to develop consolidated, lightweight components that lead to more efficient vehicles. In the medical industry, manufacturers are taking advantage of a wide range of high-strength and biocompatible 3D printing materials to customize designs and create functional prototypes, true-to-life anatomical models and surgical grade components [2].

While the advantages compared to traditional technologies are making AM more and more popular, there are still considerable improvements to be made in terms of process repeatability and part qualification before AM, and metal PBF in particular, can be applied to mass production. Due to the complex interaction of many machine and process parameters at different scales, the layering process generates a variety of defects that may compromise the geometrical and mechanical properties of the fabricated object, making expensive and time-consuming post-process qualification analysis a necessary step for most applications.

To overcome this problem, in the last few years there are growing efforts towards the design of monitoring systems, to detect or even correct the arising defects at an early stage during the layering. The process information can be monitored *in-situ* by a multiplicity of heterogeneous sensors and cameras embedded into the AM machine, providing a very large stream of data that can be effectively used to characterise the fabrication quality [3], [4], [5]. The collection and interpretation of these data opens the way not only to the early detection of many defects, but also to a fine control of the process to avoid the generation of defects altogether. In this context, data analytics approaches, mostly combining Machine Learning (ML) methods with other approaches targeting specific types of sensors (e.g., Computer Vision and signal processing) are playing a very important role [6].

Nonetheless, while the possibility to generate large amounts of sensing data is opening new unforeseen possibilities, to this date the application of data-driven monitoring approaches in real industrial settings is still limited by major issues related to the efficient retrieval, storage, visualisation and systematic analysis of the data [7]. In this context, several challenges can be highlighted:

1) *AM system HW/SW monitoring equipment.* To this date, most industrial AM systems do not provide the hardware and software equipment to assess the quality of the fabricated products during the layering. In the standard cases, even though multiple sensors are present in the machines, data interpretation is very limited (for example, based on fixed empirical thresholds on individual sensors triggering warning or error messages), and the raw sensed data is not made available to the users. In recent years, more companies are providing commercial software for real-time visualisation and monitoring of a number process parameters. Nonetheless, these solutions are generally limited in their scope, as they do not develop a fully automated defect detection strategy, failing to even detect minor defects in the printed part that could be easily corrected before another layer is built [8].
2) *Heterogeneous data integration and storage.* A single AM build can potentially generate huge amounts of heterogeneous data (in the order of terabytes, for a standard metal PBF process). To fully characterise a process, useful information may include not only the output of heterogeneous sensors and cameras mounted in the machine, but also different sources like machine parameters and settings, powder characteristics, design information such as CAD models and STL files, as well as post-processing tests outcomes. These data may be either structured (e.g., images acquired by in-situ cameras) or unstructured (e.g., reports of post-processing analysis) and have different formats and granularity. This, added to the lack of a common reference data structure and of Application Programming Interfaces (APIs) for accessing and handling the information from different sources, poses huge challenges to data collection, storage, synchronisation and correlation [9].
3) *Training data collection and annotation.* Recent literature shows the many potentials of using data-driven approaches (either standard Machine Learning, Artificial Neural Networks or Deep Learning) to the monitoring and control of different types of AM processes, including metal PBF. Nonetheless, ML typically requires the availability of large amount of training data, which is often unfeasible in a real industrial setting due to difficulties and costs of storing and appropriately annotating the training samples [10]. On these premises, even though they show interesting results, most of the available literature approaches are typically no more than proofs-of-concept, very difficult to replicate in a real industrial setting and with very limited scope. Moreover, the lack of public benchmarks and/or datasets makes it very difficult to effectively assess the accuracy and generalisation capabilities of the proposed solutions.

In this work, we deal with the problem of in-situ defect monitoring of metal PBF, taking all the challenges mentioned above into account.

We start by analysing the AM fabrication process from design to post-processing, to provide a clear overview of the data involved, and of their corresponding structure and characteristics. Then, we propose a scheme of distributed AM monitoring platform tackling all these data. To ensure the flexibility and scalability of the system, our platform is designed with a microservice software design pattern, with discrete services implementing specific functions which can be built, deployed and scaled independently. Each microservice communicates with the others through standardised APIs and protocols, enabling the integration of different technologies and the addition of new components in a plug-and-play fashion.

On these premises, we deal with the specific problem of in-situ PBF defect detection as a plugin component of our distributed microservices platform, entailing all the phases of the process. More specifically, we focus on the real-time detection of powder bed defects with image analysis,

exploiting an off-axis low-cost camera mounted on top of a DLMS machine. The defect detection framework is fully implemented and exploited in a real industrial DMLS machine operating in Stellantis automotive company. Our proposed solution consists of a set of fully-automated Computer Vision and Machine Learning techniques to detect five different categories of powder bed defects that are hard to spot by visual inspection and to monitor the profile of the fabricated objects in real-time. By allowing the early stopping/correction of the faulty objects, this system is expected to improve process repeatability and majorly reduce human intervention and the necessity of post-process analysis, with major positive impacts on the production costs. To address the problem of the lack of annotated images, we add to the framework an additional synthetic image generation module, that uses ML, and more specifically Generative Adversarial Network (GAN), to generate realistic images with defects similar to the ones occurring during a typical PBF build.

The rest of the paper is structured as follows. Section 2 provides an overview of the state of the art of the monitoring systems for AM, with specific regards with AM data collection/integration and powder bed defect detection based on imaging data. Section 3 describes our proposed solution, including the scheme of our microservices AM monitoring platform, along with the specific modules for PBF defects detection, part profile monitoring and synthetic image generation. In Section 4, we provide and discuss experimental results from our real industrial case-study in Stellantis. The paper is concluded in Section 5.

2 Background

With the growth of industry 4.0 and increasing role of AM across multiple sectors, future developments are expected to lead to a closer interrelationship between AM and the other Industry 4.0 pillars, with special regards with Internet of Things (IoT), cloud computing and big data analytics [11]. In this context, the development of cloud-based platforms for AM process monitoring, with IoT devices and machines communicating with each other, becomes fundamental to fulfil the potentials for mass production. This is indeed a very hot problem, that is destined to attract ever-increasing attention from the research community. In [12], the authors analyse the benefits of integrating IoT devices with the AM process, coming to the conclusion that cloud computing and IoT can improve the efficiency of the AM processes and reduce manufacturing waste while fulfilling customer specifications. In [13], they highlight the limitations of the current cloud-based platforms, which fail in supporting the customers throughout the AM process, and propose a solution supporting remote printing as well as design and process planning.

Due to the strong motivation to meet industrial requirements in terms of product quality and reduced post-processing analysis, even the specific problem of in-situ AM monitoring, which is poorly addressed by commercial solutions [14], [15], is attracting more and more attention in literature. In the last few years, researchers are applying more and more data-driven approaches, with specific regards with ML methods, to different types of sensed data, to obtain automated detection of a number of defects in AM fabricated objects [16], [17], [18]. Nonetheless, the problem of in-situ monitoring is mostly addressed as if it was independent from the one of AM heterogeneous data collection/integration, which considerably limits the scope and real applicability of the proposed solutions. Extensive research is still needed to integrate the in-situ monitoring techniques within a standardized *digital thread* for AM, where ML algorithms can directly access and create meaningful correlation between design, process parameters and post-process analysis.

Among the proposed monitoring solutions, visual, camera-based methods have been extensively used to identify PBF processing errors, such as powder bed discontinuities and geometrical defects [19]. Nonetheless, a typical problem is the lack of generalisation, as the proposed solutions are ad-hoc for specific machines and experimental set-ups. Other solutions exploit images acquired by high-speed cameras to analyse the characteristics of the by-products and of the laser-induced vapour plume of Laser Powder Bed Fusion (LPBF) during the powder melting, based on the hypothesis that these characteristics can be correlated with micro-structural and mechanical defects of the fabricated objects [19], [20]. Nonetheless, these solutions are majorly affected by the disturbance caused by vapour plume ejection and they require expensive equipment that is typically not available in a standard industrial setting. Finally, a lower number of solutions exploit low-cost thermal imaging, relying on relative temperature measurements to detect unwanted process variability, even though the proposed techniques are not real-time [21].

In general, as already mentioned in Section 1, a common limitation of all the data-driven techniques is that they typically require to be trained on a large amount of annotated data (e.g., for image-based systems, a large amount of images of the powder bed per each possible defect and condition). As the intentional generation of defective parts is not viable in industry, as it is costly and time-consuming, the availability of suitable datasets for training and validation purposes becomes a critical issue. A promising and yet mostly unexplored solution to this problem, at least for camera-based monitoring systems, is the use of Generative Adversarial Network (GAN [22]), a generative model characterised by training a pair of neural networks (a generator and a discriminator) in competition with each other to create realistic synthetic images. While GANs are becoming very popular in many Computer Vision tasks, the application to the AM sector is still extremely limited. In [23], the authors analyse the benefits and challenges of GANs to generate new data for ML algorithms. In [24], the authors apply a Conditional GAN (CGAN), an extended form of GAN [25], to produce synthetic data using images from a near-infrared high-resolution optical camera, obtaining quite promising results. In our work, a similar approach is applied to a more complex task involving a large number of PBF defects, using a standard low-cost visible-range camera for image acquisition. For this purpose our solution exploits a ConSinGAN [26], an improved architecture that requires a single real powder bed image for the training.

3 In-situ monitoring framework

In this Section we describe our contributions. We first provide an overview of the microservices AM monitoring platform, with corresponding data structure. Then, we go in depth into the specific application of camera-based in-situ monitoring of powder defects, describing our solutions for layer-wise detection of powder bed defects, object profile monitoring and synthetic image generation.

3.1 Data structure and AM monitoring platform

For the development of an AM monitoring infrastructure, it is important to understand the source, type and structure of the data to be be stored and their role in the AM process. To do so, a typical AM pipeline was analysed from CAD generation to post processing analysis, with special regards with laser PBF processes. All the involved data are divided into five categories based on the life-cycle process of Fig. 1(a): i) design parameters, ii) process parameters, iii) process characteristics, iv) post-processing treatments and v) product analysis. Table 1 reports a list of data in each category.

1) *Design parameters.* Design parameters refer to all the parameters defined during the design of the product to be printed, in the first phase of the AM pipeline (see Fig. 1(a)). These parameters are of particular importance for the correct build of the object and its quality. In this step, the data range from the design features, including the CAD file and material of the produced part, to design constrains like the required surface roughness or dimensional tolerances. In this context, also the position of the supports plays a fundamental role and the data regarding the type of supports (geometry and shape), the build position and the build orientation need to be stored.
2) *Process parameters.* Process parameters involve the input settings selected during the setup of the machine. They are machine and technology dependent and they are mainly set at the start of the process and typically remain constant throughout the build. The process parameters can be related to the material used, such as the material type, composition, melting point, heat capacity, etc., or to the powder bed composition like powder density, thermal conductivity, temperature etc.. Other process parameters may include the machine settings like the type of laser, its operational mode or frequency, speed and power, the build environment gas type, level and characteristics or the recoater properties such as speed, layer thickness, powder diffusion etc..
3) *Process characteristics.* They refer to the data that can be used to characterise the printing process during the build, either acquired as-is or derived by sensors mounted on the machine. They are highly influenced by the process parameters and their number/nature depend on the specific sensing equipment of the AM system. For a laser PBF process, process parameters can be obtained at different levels of observation, from the *melt pool* (the small pool of metal liquid that is generated when the laser heats the powder) up to the whole powder bed layer. Data include temperature measures and geometrical characteristics obtained with thermal cameras, by-product measurements obtained from IR imaging like spattering and plume, laser track characteristics as well as whole layer properties including surface topography, temperature, etc. Process characteristics at this level involve also visible-range images acquired to characterise the morphological and geometrical characteristics of the powder bed as well as of the printed object.
4) *Post-processing treatments.* Post-processing treatments refers to the post-build processes required to obtain the final product characteristics. The post-processes could change based on the specific design requirements and treatment type. For example, the post-process heat treatment data, used to achieve the desired hardening or softening of a material, involve technique, temperature, atmosphere, heating and cooling time utilised. Other possible treatments are post machining (involving type of operation, velocity, tool material and geometry etc.) and powder remove treatments characterised by vibration techniques, time, etc..
5) *Product analysis.* Product analysis refers to the final quality tests of the AM fabricated parts, performed to verify that all the characteristics of the object obtained after the post-processing treatments comply to the design requirements as well as to the industry standards. The outcomes of these tests define or characterise the success of the build. Hence, they are of fundamental importance for future improvement of the product quality, for design optimisation as well as for the training of supervised ML models. In this context, data include dimensional and geometrical accuracy, surface roughness, porosity, mechanical properties, chemical properties, etc..

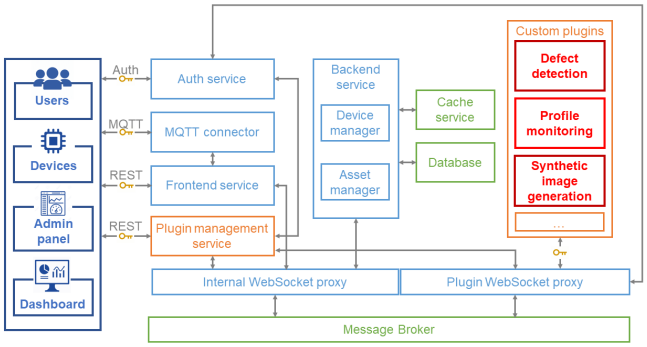


Fig. 2: AM monitoring platform: proposed architecture.

All the categories previously described involve different types of data (textual, numeric, CAD, images, reports, etc.) that need to be monitored, exchanged between different assets and finally stored. To do so, we propose the AM distributed monitoring platform scheme reported in Fig. 2, that is an extension and customization of a general-purpose open-source industrial framework [27]. As already mentioned in Section 1, to ensure the flexibility, modularity and scalability of the platform, the infrastructure is composed of several microservices components that allow to: i) manage different users and devices, ii) receive, monitor, and store the data available, iii) provide API for internal and exter-

TABLE 1: Main data in the Additive Manufacturing process

Category	Type of data
Design parameters	Design features (CAD, Part material, Roughness, etc.) Build features (Supports, build position and orientation, etc.)
Process parameters	Laser parameters (Type, Operational mode, Wavelength, Frequency, Width, Power, Scan speed and pattern, etc.) Material properties (Type, Heat capacity, Conductivity, Melting temperature, Composition, Particle structure, etc.) Powder bed properties (Powder bed density, Thermal conductivity, Heat capacity, Temperature etc.) Recoater properties (Type, Speed, Powder diffusion, Layer thickness, etc.) Build environment properties (Gas type, level and characteristics, pressure, temperature etc.)
Process characteristics	Melt pool (Temperature and geometry, etc.) By-products (Plume, Spatter, etc.) Laser track (Width, Continuity, Depth, Defects, etc.) Layer (Surface topography, Temperature, Defects, etc.)
Post-processing treatments	Heat treatment (Technique, Temperature, Atmosphere, Heating and cooling time, etc.) Post Machining (Machining operation, Velocity, Feed rate, Depth of cut, Tool material and geometry, Environment, etc.) Powder remove treatment (Vibration settings, Time etc.) Other process (Painting, Engraving, Process characteristics etc.)
Product quality analysis	Geometry and dimension analysis (Accuracy, Precision, etc.) Surface quality (Roughness, Deformation, Defects, etc.) Physical properties (Density, Porosity, Microstructure, etc.) Mechanical properties (Yield and Tensile strength, Elongation, Fatigue, Hardness, etc.)

nal communication, even with third-party software and iv) integrate new functionalities (e.g. in-situ defects detection, predictive maintenance, etc.) through plug-and-play components.

First of all, the *Message Broker* together with the *Internal* and *Plugin WebSocket proxy* are in charge of managing all the communication between the platform components, using several protocols like Message Queuing Telemetry Transport (MQTT) and Representational State Transfer (REST) web services. MQTT is a communication protocol that implements the publish/subscribe paradigm [28], while REST [29] is a software architectural style implementing request/response. REST is commonly used to create interactive applications that use Web services.

The *Frontend service* and *MQTT connector* manage the communication between the platform and the external assets. The Frontend service provides access to the resources through a Dashboard and allows the management of the users using an Admin panel and the REST protocol, while the MQTT connector allows the communication between the platform and the devices using the MQTT protocol. The *Auth service* is in charge of managing the authentication of the users and devices in order to secure the data transferred and stored.

The *Backend service* is used to internally manage the databases and devices available using two modules: the *Asset Manager* and the *Device Manager*. The Asset Manager is a software module that handles all the information regarding users and devices (for example, the user authorisation, the process data or the device location). The Device Manager provides the connection between the assets, the database and the plugins that require to store or retrieve the data available.

The *Device module* provides a crucial component called *Device Connector* to integrate heterogeneous IoT devices into the platform, providing functionalities to: i) provision new devices exploiting the Asset Manager; ii) enable a secure and trusted communication through the Auth Service module; iii) allow a bidirectional communication in MQTT and REST thought the MQTT Connector and Frontend service, respectively; iv) send and store new sampled data in the remote database via the Device Manager; v) execute plugins on edge and manage their communications with the Plugin Management Service.

Finally, the *Custom Plugins* are the additional components that can be added to the system in a plug-and-play fashion. They are integrated in the platform using the Plugin management service that allows the communication with both internal and external resources, besides providing the data required. The plugins can implement different functionalities that are typical of an Industry 4.0 setting: for example, anomaly detection, predictive maintenance, or in the specific case-study of this work in-situ defect detection, part profile monitoring, etc.

The proposed AM distributed monitoring infrastructure can be developed in cloud systems and offered to companies as Platform as a Service (PaaS) [30]. PaaS is a category of cloud computing services that provides to the customers a platform to build, run, and manage applications without the need of building and maintaining the infrastructure. PaaS solutions can be alternatively provided as public, hybrid or private cloud, thus, reducing the data exchange over the Internet and increasing the privacy and data management. Thanks to these features provided by the distributed platform, plugins can be developed to be either executed remotely on the cloud or locally on the edge.

In the next part of this Section, we will focus on the specific application targeted by our work, that is the in-situ monitoring of powder bed. This includes three functionalities (and corresponding custom plugins, that are highlighted in red in the scheme of Fig. 2): the automated detection of of the powder bed defects, the real-time monitoring of the object profile during the layering as well as the automated generation of synthetic powder bed images.

3.2 Powder bed monitoring equipment

As anticipated in Section 1, PBF is a discrete process where a mechanical recoater spreads a layer of powder on top of a build plate. Then, a directed laser selectively melts the powder, which consolidates and bonds to the previous layer. To fully monitor the evolution of this process, the system needs to acquire images of the build plate on a layer-by-layer basis, respectively before and after the action of the laser. This is a functionality that is not inbuilt in the standard AM machine used in our study (EOS M290 DMLS

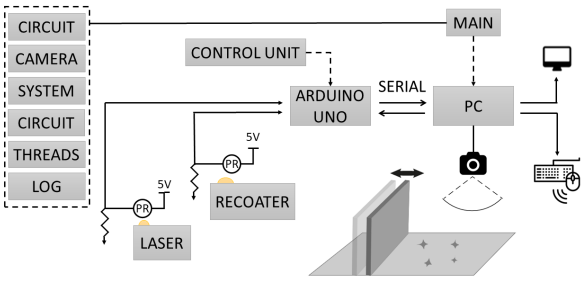


Fig. 3: In-situ camera-based monitoring equipment.

printer). Then, we set-up an acquisition equipment using low-cost hardware, as shown in Fig. 3, which implements the functionalities of the *Device Connector* (see Section 3.1). It consists of:

(a) an Arduino Uno microcontroller board directly connected with the AM machine, used to operate the camera, trigger the acquisition and take images of the build plate based on the AM machine signals.
(b) an IDS UI-1540-SE 1.31Mpix camera (1280×1024 resolution). The camera is controlled by the API provided by the manufacturer. In order to avoid any interference, the camera is mounted on top of the machine, off-axis with respect to the optical path of the laser.
(c) An embedded PC running Linux (e.g. Raspberry PI), to temporarily receive the acquired images, send them thought REST to the Database module of the monitoring platform, and optionally run services on the edge (for example, CV and ML algorithms).

The light signals emitted respectively by the laser and recoater operations, detected by means of photo-resistors, are used to trigger the image acquisition before and after the laser action, without requiring any input from the user.

3.3 Defects detection

The powder bed defect detection module is composed by a set of image analytics algorithms, written in Python using OpenCV and Keras standard libraries for, respectively, image processing and ML computation. The system allows the real-time detection of five different categories of powder bed defects shown in Fig. 4 (a)-(e), that are known to translate into porosity or micro-structural alterations of the AM fabricated object which may be a cause of discard. Hence, it is important that they are detected at the earliest fabrication stages:

(a) *Holes*: localised lacks of metallic powder that create small dark areas in the powder bed image. They are generally due to a bad regulation of the powder dosing factor, leading to local lacks of powder.
(b) *Spattering*: droplets of melted metal ejected from the melt pool and landed in the surroundings.
(c) *Incandescence*: high-intensity areas in the powder bed layer. It is generally a consequence of the inability of the melt pool to cool down correctly, due to an excess of laser energy power.
(d) *Horizontal* defects: dark horizontal lines in the layer image caused by geometric imperfection of the piece that leads to the incorrect spreading of the metallic powder.
(e) *Vertical* defects: vertical undulation of the powder bed along the direction of the recoater's path, consisting in alternated dark and light lines. The origin is either a mechanical defect of the recoater's surface or a mechanical interference between the object and the recoater.

Real-time defects detection is composed of several image processing steps.

(a) *Normalization.* In order to correct uneven illumination problems, the images are normalised against a common reference frame. To do so, the system uses as a reference an image of the powder bed acquired before the start of the print process.
(b) *Contrast enhancement.* To make the objects more identifiable and enhance the contrast against the background, a standard background subtraction algorithm is applied [31].
(c) *Objects identification.* Automated intensity thresholding is applied to obtain a rough identification of the printed objects against the background, as well as to identify intensity discontinuities due to powder bed inhomogeneity.
(d) *Morphological filtering.* Specific objects of interest are identified based on their shape, exploiting morphological algorithms. More in detail, Watersheds and Hough transforms, followed by standard morphological regularization (i.e., opening, closing, holes filling) are applied to identify, respectively, round-shaped and horizontal/vertical lines in the images. Based on the specific shape and number of objects, the software is able to determine whether there is a defect, and to associate the image to a specific defect category.

As an example, Fig. 5 shows the intermediate results of the image processing pipeline, applied to an image with a spattering defect (i.e., tiny particles of liquid metal being ejected from the laser's path). This is one of the most frequent defects occurring during laser PBF, which may spoil the powder bed and generate increased porosity, roughness, and lack of adhesion in the finished parts. Fig. 5(d) shows the metal spatters, as identified by the algorithm after the morphological filtering step.

3.4 Profile monitoring

Besides powder bed defects detection, our solution provides the possibility to monitor the layer-wise profile of the printed parts in real-time (see an example in Fig. 4(f)), in order to detect profile alterations during the build.

Given the variability of shape and dimension of the printed parts, this operation is computationally more demanding compared to basic powder bed defects detection, and can be addressed as a semantic segmentation task. Semantic segmentation is a specific Computer Vision task that partitions an image into portions belonging to the same object, by associating an object-specific label to each pixel. In other words, it is an image segmentation task addressed as a binary classification task at the pixel-level, with two classes respectively representing an object of interest (in our case, the fabricated part) and the background.

For this purpose, our system deploys a U-Net architecture [32], that is a state-of-the-art deep learning network for semantic segmentation originally presented for biomedical

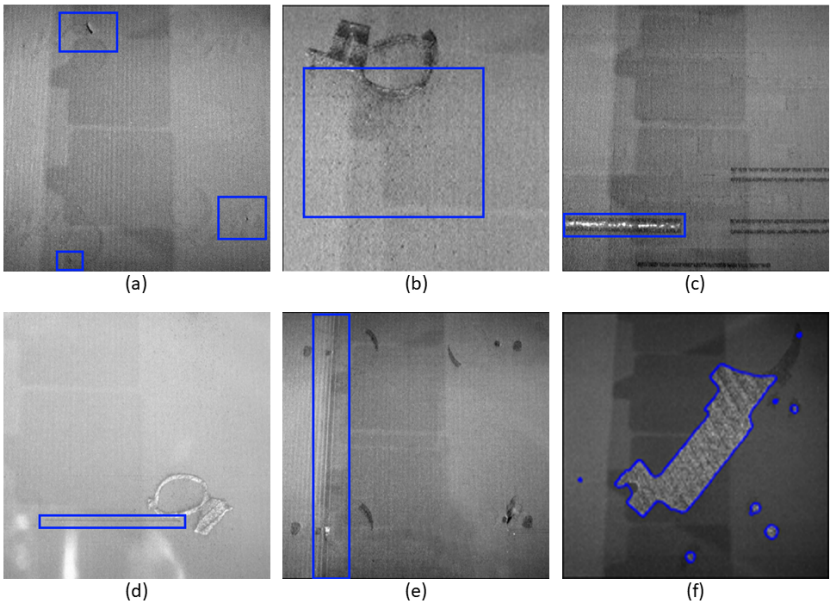


Fig. 4: Examples of powder bed monitoring targets. (a) Holes defect, (b) Spattering defect, (c) Incandescence defect, (d) Horizontal defect, (e) Vertical defect, (f) Printed parts' profile monitoring

image segmentation and then successfully applied across multiple CV applications. The algorithm implements an end-to-end fully convolutional network (FCN) consisting of convolutional and pooling layers without any dense layer, which makes it fit for any image size.

As shown in Fig. 6, the architecture is composed of two paths. The first is the encoder or contraction path, which provides an understanding of the context in the image, and consists of several convolutional and max-pooling layers with gradually decreasing feature map dimension. The second is the decoder or symmetric expanding path,

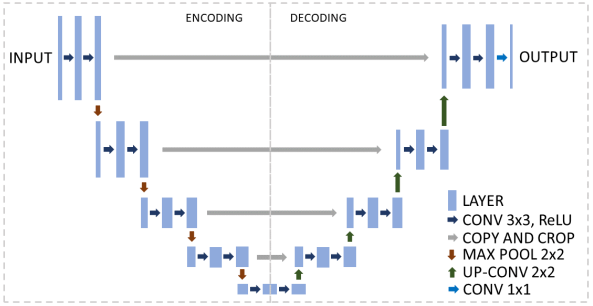


Fig. 6: U-Net architecture.

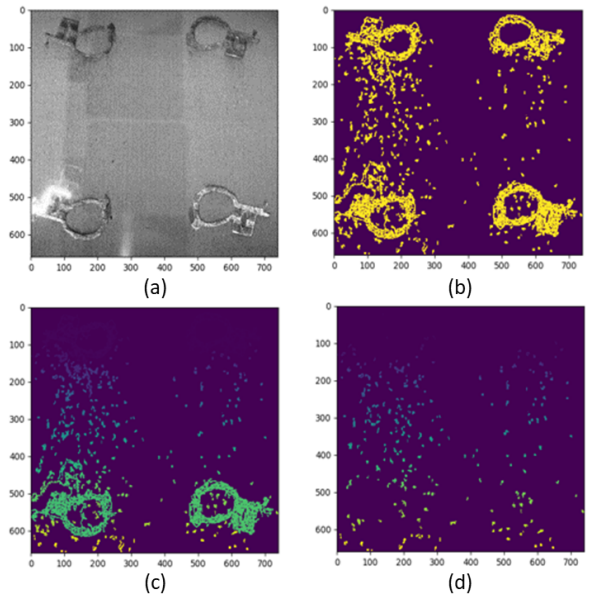


Fig. 5: Example of spattering defects detection pipeline: (a) Normalization, (b) Contrast enhancement, (c) Objects identification, (d) Morphological filtering.

which provides precise object localization using transposed convolutions.

In our approach, the U-Net is initialized as reported by [32] and then fine-tuned on a representative PBF layer image, acquired by our infrastructure.

3.5 Synthetic image generation

As already anticipated, a typical challenge of supervised ML algorithms is the need for a large number of annotated training images. For the specific task of defect detection, training images should ideally represent all possible targeted classes, with and without defects. The availability and annotation of such images is typically an issue in a real industrial setting, especially for the defect categories.

To overcome this problem and augment the training dataset that is available to our ML services, our system exploits ConSinGAN (Conditional Single-Image Generative Adversarial Network [26]), an extended form of Generative Adversarial Network that allows to generate realistic synthetic data starting from a single training image. This is a very favourable characteristic for our specific task, where the option of intentionally generating defective parts just for ML training purpose is ruled out.

As shown in Fig. 7(a), a classic GAN architecture is composed by two Neural Networks, the Generator and the Discriminator, that compete to develop a generative model. Starting from an input image, the generator is trained to produce synthetic data that resemble the real available samples. At the same time, the discriminator network, a binary classifier, is trained to discern the synthetic samples produced by the generator from the real ones. In the training phase, the two networks compete with each other, in the sense that the generator attempts to raise the error rate of the discriminator, while the discriminator tries to improve at discerning synthetic and real data. Hence, the idea is to realise an indirect training of the generative model through the discriminator: no real data are accessible by the generator, and it can only interact with the discriminator [33].

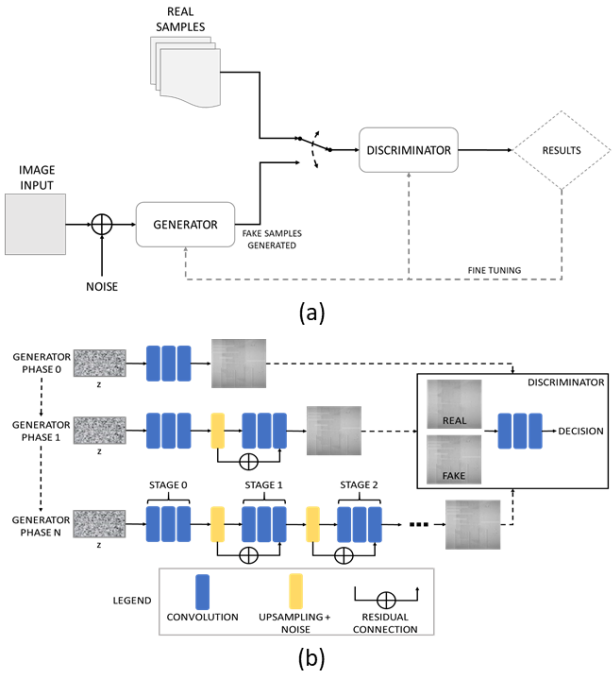


Fig. 7: Generative Adversarial Network. (a) Traditional scheme. (b) ConSinGAN architecture.

The ConSinGAN is a GAN built with a multi-stage and multi-resolution approach, as shown in Fig. 7(b). First, as input for few iterations, an image with a coarse resolution is used for the ConSinGAN to learn mapping a random noise vector z to a low-resolution image (See *Generator Phase 0* in Fig. 7(b)). After the initial training has converged, the size of the generator is increased, adding three additional convolutional layers. Each stage takes the raw features from the previous stage as input, and a residual connection is used to feedforward the last convolutional layer (see *Stage 1* in Fig. 7(b)). The process is iterated N times until the desired output resolution is achieved.

The discriminator takes in input the synthetic image created by the generator, together with a single real image from the dataset, and it is trained to decide whether the given images are real or synthetic, by comparing them.

During the synthetic image generation, the most critical part is the so-called *image harmonization*. It consists in transforming a composite image, called Naive, into a more realistic one, by applying the same style and appearance of an original image, so-called Training. The output of this process is a synthetic Harmonized image that contains the specific objects of the Naive image, but with a general appearance that is similar to the Training one (see two examples in Fig. 8, respectively for a generic natural image and for a powder bed image of our system).

In the practice, the network is first trained to learn a generative model from the original Training image. Then, when it is given the Naive image as input, it will attempt to create a synthetic image resembling the original learned distribution. Hence, once the model is learned from a single Training image, several synthetic ones can be generated, by simply changing the Naive picture given as input.

For the specific task of generating synthetic images with powder bed defects, the harmonization is implemented as follows: i) the generator is trained using a real Training image without defects (Fig. 8 (2-a)); ii) a Naive image is generated by modifying the Training one with a photo editing software. The modifications in this case may represent specific powder bed defects, for example holes and horizontal lines (Fig. 8 (2-b)); iii) the Naive image is harmonized to resemble the characteristics of the Training one (Fig. 8 (2-c)). By doing so, the original image assimilates the synthetic defects.

The synthetic images can then be made available to the platform, for data augmentation purposes.

4 Experimental Results

In the following, we assess the accuracy of our defect detection, profile monitoring and synthetic image generation algorithms, using the experimental test-case developed in Stellantis.

To validate the *defects detection* algorithms, we exploited a test set of 24 pre-annotated images representing the five powder bed defect categories targeted by our application. The goodness of the defects detection was quantified by extracting metrics that are largely employed in descriptive statistics:

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

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F\text{-}score = \frac{2 * Recall * Precision}{Recall + Precision}$$

In our application, True Positives (TP) are the outcomes when the algorithm correctly identifies a defect that was really present. True Negatives (TN) are the instances when the algorithm was right in not detecting a defect that was not present. False Positive (FP) and False Negative (FN) are the errors of the algorithms, respectively when the model detected a defect that was not present, or it was not able to identify a defect that was present.

Table 2 reports the obtained results, separately per each category of defect.

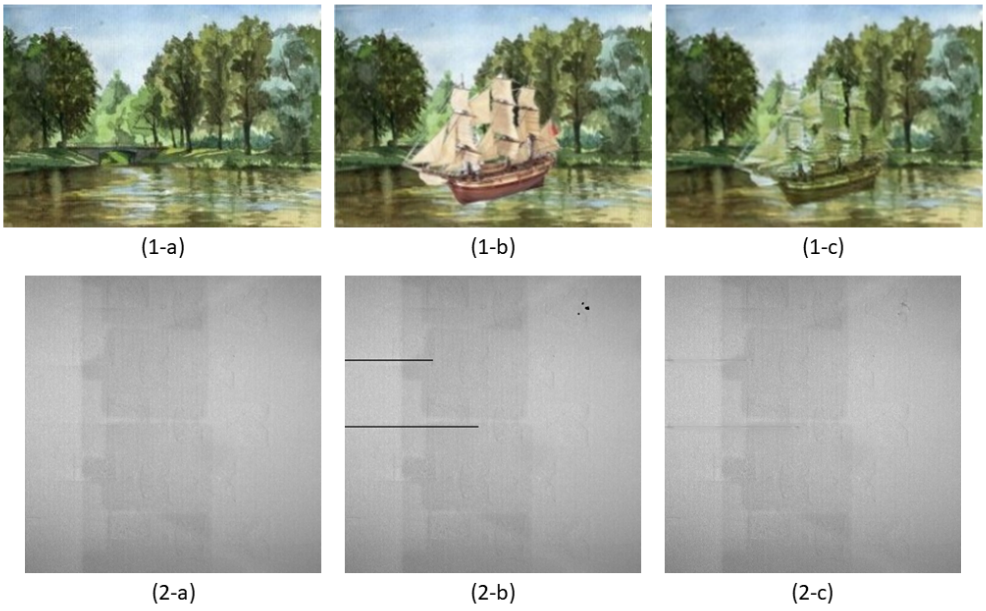


Fig. 8: Example of image harmonization with a generic image (1) from [26] and a powder bed image (2): (a) Training images, (b) Naive images and (c) Harmonized images.

TABLE 2: Defects detection validation results

	Holes	Spatt.	Incand.	Horizontal	Vertical
TP	14	22	15	11	6
TN	8	1	4	11	16
FP	2	1	5	1	0
FN	0	0	0	1	2
Accuracy	91.3%	95.8%	79.2%	91.6%	91.6%
Precision	87.5%	95.6%	75.0%	91.6%	100.0%
Recall	100.0%	100.0%	100.0%	91.6%	75.0%
F-score	0.93	0.98	0.86	0.92	0.86

As it is visible from Table 2, we obtained values $\geq 75\%$ for all the metrics, with F-score $\geq 86\%$ in all categories of defects. According to our tests, the most challenging defect is the Incandescence, probably due to the wide variability of pixel luminosity. On the other hand, Spattering defects, which are the most common defects, are also the easiest due to the high number of spatters generated.

A second test-case was implemented to validate the *profile monitoring* algorithm. In this case, as the profile monitoring is a semantic segmentation task, the validation exploited a widely used image segmentation metric, the Dice Similarity Coefficient (DSC). This metrics is used to compare an automatic segmentation against a manually obtained ground truth by computing a 0 to 1 similarity values of the two binary images, as follows:

$$DSC = \frac{2|X \cap Y|}{|X| + |Y|}$$

where $|X|$ and $|Y|$ are the number of pixels of the two images (respectively, the automatic segmentation and the ground truth) and $|X \cap Y|$ is the number of pixels that are in common to the two images. Fig. 9 shows and example, where (a) is the binary mask obtained by manual segmentation, used as the ground truth, and (b) is the binary mask obtained by our profile monitoring suite.

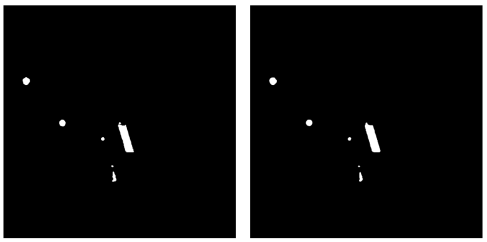


Fig. 9: Profile monitoring validation: (a) Ground Truth (b) Automatic Algorithm.

Our experiment was run on 44 independent profiles, obtained from 3 different AM parts at different layering stages. As a result of this experiment, we obtained a mean DSC value equal to 0.927, when considering the DSC of each single object in a layer image, and to 0.953, when considering the DSC value of a layer image taken as a whole. In both cases, the obtained results confirmed a very high similarity between automatic segmentation and ground truth.

To ensure a real-time response, both defect detection and profile monitoring need to provide results before the next layer is started, meaning that the execution time needs to be lower than the time elapsing between two subsequent layers. For a standard laser PBF process targeted by our system, this time is in the order of tens of seconds. To assess the real-time capability, the execution time of both the tested algorithms was computed for both detection and profile monitoring tasks, as reported in TABLE 3.

As shown in the table, the obtained execution times are all $\leq 2.5s$, which is well below the given acceptance threshold. The algorithm requiring the longest execution time ($2.461s$) is the profile monitoring, which implements

TABLE 3: Mean execution time of the algorithms.

Operation	Time [s]	Operation	Time [s]
Holes	0.791	**Horizontal**	0.593
Spattering	0.574	**Vertical**	0.932
Incandescence	0.821	**Profile monitoring**	2.461

a deep neural network. All the other algorithms exploiting standard image processing operations have execution times $\leq 1s$.

The final test-case involves the *synthetic image generation* algorithm. Assessing the validity of synthetic data generation is indeed very challenging, as there is not a standard quantitative protocol to follow, nor an absolute figure of merit to base the evaluation on. In our test-case, we used the Fréchet Inception Distance (FID) [34], which defines the statistical distance between feature vectors calculated for real and generated images:

$$FID = ||\mu_X - \mu_Y||^2 + Tr(\Sigma_X + \Sigma_Y - 2\sqrt{\Sigma_X \Sigma_Y})$$

where X is the set of real images; Y is the set of synthetic images; μ_X and μ_Y are the feature-wise mean of the real and generated images, respectively; Σ_X and Σ_Y are the covariance matrix for the real and generated feature vectors, respectively; Tr refers to the trace operation in matrix, which is the sum of the diagonal elements; $||\mu_X - \mu_Y||^2$ refers to sum squared difference between the two mean vectors. The FID value varies between 0 and plus infinite, respectively corresponding to two images that are the very same and to two images that are completely different.

In our experiment, we artificially generated 35 synthetic images with combinations of three types of defects: holes, vertical and horizontal, respectively. Fig. 10 shows the progress of the FID metric of the ConSinGAN generator during the training phase, reporting the mean FID value every 100 steps. As it can be seen from the plot, FID followed a decreasing trend with minor fluctuations, meaning that the network progressively improves in the generation of images as the training proceeds, and generates synthetic images that resemble the real ones more and more. More specifically, FID values reduced from a starting mean value of 303.80 at step 1, to a final mean value of 118.40 after 2000 steps.

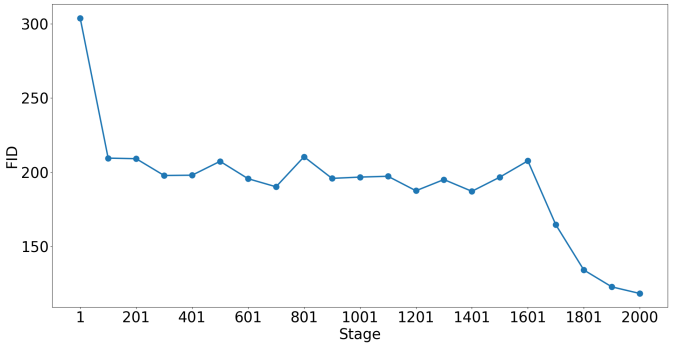


Fig. 10: FID trend on generated images during the training phase.

Unfortunately, FID metric provides an abstract measure which may return very different values based on the specific content and visual characteristics of an image. Hence, there is not a specific FID value that can be deemed universally acceptable in every domain, and the obtained results need to be adequately interpreted based on the specific application. To provide a qualitative figure of merit for our experiments, in Fig. 11 we show an example of a real image without defects (a) and a synthetic image (b) with defects, as generated after 2000 steps with our approach. As it can be seen from the figure, the two images are very similar to each other. This corresponded to a FID value of 89.86, meaning that values in the order of one-hundred represent a very good outcome in this specific application.

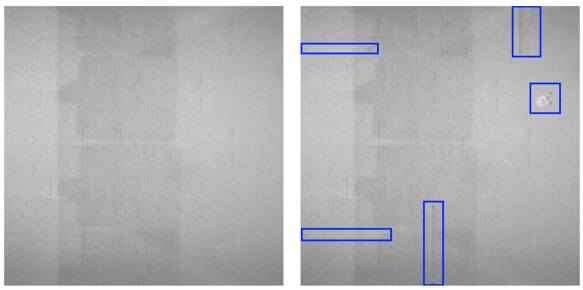


Fig. 11: Comparison between a training image (a) and a synthetic one (b) with defects (holes, horizontal and vertical lines).

The FID results obtained on the test-set of 35 synthetic images are reported in TABLE 4, grouping the images in terms of the combination of defects they show, which all included a very wide variety in terms of number and size of the single defects. As a reflection of this variability, as expected the FID values varied from very low values $(10 - 20)$ in images with fewer small-sized defects (holes, especially), to higher values $(200 - 290)$, in images with very large number of big-sized defects. In all cases, the FID values followed a decreasing trend during the image generation process, demonstrating that the final similarity to real images improves with the training and that the image generation algorithm is stable.

TABLE 4: ConSinGAN results

Image	Type of defect	FID	Image	Type of defect	FID
1	Horizontal	182.73	21	Horizontal + Vertical	89.86
2		122.10	22		195.02
3		48.93	23		287.23
4		53.25	24		159.56
5		319.77	25	Holes	97.74
6	Vertical	129.66	26	Horizontal + Holes	52.11
7		45.64	27		141.67
8		20.14	28		49.22
9		269.04	29		150.01
10		201.72	30		62.01
11	Holes	15.89	31	Vertical + Holes	81.61
12		11.75	32		134.80
13		78.95	33		160.73
14		65.35	34		85.50
15		71.91	35		291.12
16	Horizontal + Vertical	57.53			
17		124.74			
18		145.17			
19		75.07			
20		66.40			

As a final experiment to provide a quantitative demonstration the validity of the synthetic images, we run our

defect detection algorithm on them. The ratio of this experiment is that, if we take into account its inherent error rate, the defect detection algorithm can be used as indirect measure of how similar the images are to real PBF defects. TABLE 5 reports the defect detection results obtained on the test-set of 35 synthetic images with holes, vertical and horizontal defects, generated with the ConSinGAN.

TABLE 5: Defects detection results on the synthetic images.

	Holes	Horizontal	Vertical
TP	16	15	14
TN	12	14	15
FP	3	1	0
FN	4	5	6
Accuracy	80.0%	82.9%	82.9%
Precision	84.2%	93.8%	100.0%
Recall	80.0%	75.0%	70.0%
F-score	0.82	0.83	0.82

Overall, while the metrics are slightly worsened compared to the ones that were obtained on the real images (see TABLE 2), this difference was reasonably low. As an example, the F-score was lower for all categories of defects in the synthetic images (Holes: 0.82, Horizontal: 0.83, Vertical: 0.82) with respect to the real ones (Holes: 0.93, Horizontal: 0.92, Vertical: 0.86), with a maximum difference of 0.11 for the holes defect. While this difference is reasonably due to the fact that the synthetic images were intentionally generated with a very large number of defects to challenge the algorithm, the acceptable performance of the defect detection experiment still demonstrates a good level of similarity between the real and the synthetic dataset.

5 Conclusions and Future Work

This paper proposed a low-cost camera-based in-situ monitoring system for metal PBF that provides the real-time detection of five different defects as well as the layerwise monitoring of the printed part profile. For dataset augmentation purposes, the system even allows to generate synthetic images offline with GAN. Tackling the problem of AM monitoring with a holistic view, all these functionalities are embedded into an integrated distributed software infrastructure for AM, that allows the collection, integration and storage of data at all stages of the AM pipeline. Such distributed infrastructure has been designed and developed following the microservice software design pattern, ensuring the modularity, flexibility and scalability. It also allows the integration of heterogeneous technologies and devices, as well as the extension with new industry 4.0 functionalities in a plug-and-play fashion.

Our camera-based system is already operational in Stellantis automotive company. While the presented case-study only deals with visual imaging of powder bed defects, the monitoring framework is already designed in such a way that other types of sensors can be easily integrated. More specifically, our future work will address Acoustic Emission (AE) data, which has been demonstrated to be an effective and economical solution to predict several issues including fracture, plastic deformation, and crack initiation and growth [35]. While the integration of visible-range imaging and AE opens the way to improved joint monitoring of macroscopic and structural integrity defects, it also requires extensive efforts towards enhanced techniques for multi-modal data fusion, which is still a very open problem [36].

References

[1] N. Guo and M. C. Leu, "Additive manufacturing: technology, applications and research needs," *Frontiers of Mechanical Engineering*, vol. 8, no. 3, pp. 215–243, 2013.

[2] M. Salmi, K.-S. Paloheimo, J. Tuomi, J. Wolff, and A. Mäkitie, "Accuracy of medical models made by additive manufacturing (rapid manufacturing)," *Journal of Cranio-Maxillofacial Surgery*, vol. 41, no. 7, pp. 603–609, 2013.

[3] J. L. McNeil, K. Sisco, C. Frederick, M. Massey, K. Carver, F. List, C. Qiu, M. Mader, S. Sundarraj, and S. Babu, "In-situ monitoring for defect identification in nickel alloy complex geometries fabricated by l-pbf additive manufacturing," *Metallurgical and Materials Transactions A*, vol. 51, no. 12, pp. 6528–6545, 2020.

[4] M. Montazeri, A. R. Nassar, A. J. Dunbar, and P. Rao, "In-process monitoring of porosity in additive manufacturing using optical emission spectroscopy," *IISE Transactions*, vol. 52, no. 5, pp. 500–515, 2020.

[5] H. Rezaeifar and M. Elbestawi, "On-line melt pool temperature control in l-pbf additive manufacturing," *The International Journal of Advanced Manufacturing Technology*, vol. 112, no. 9, pp. 2789–2804, 2021.

[6] L. Meng, B. McWilliams, W. Jarosinski, H.-Y. Park, Y.-G. Jung, J. Lee, and J. Zhang, "Machine learning in additive manufacturing: A review," *JOM*, pp. 1–15, 2020.

[7] S. Di Cataldo, S. Vinco, G. Urgese, F. Calignano, E. Ficarra, A. Macii, and E. Macii, "Optimizing quality inspection and control in powder bed metal additive manufacturing: Challenges and research directions," *Proceedings of the IEEE*, vol. 109, no. 4, pp. 326–346, 2021.

[8] V. Carl, "Monitoring system for the quality assessment in additive manufacturing," in *AIP Conference Proceedings*, vol. 1650, no. 1. American Institute of Physics, 2015, pp. 171–176.

[9] A. Simeone, A. Caggiano, and Y. Zeng, "Smart cloud manufacturing platform for resource efficiency improvement of additive manufacturing services," *Procedia CIRP*, vol. 88, pp. 387–392, 2020.

[10] A. Z. A. Kadir, Y. Yusof, and M. S. Wahab, "Additive manufacturing cost estimation models—a classification review," *The International Journal of Advanced Manufacturing Technology*, vol. 107, no. 9, pp. 4033–4053, 2020.

[11] J. Butt, "Exploring the interrelationship between additive manufacturing and industry 4.0," *Designs*, vol. 4, no. 2, p. 13, 2020.

[12] R. Ashima, A. Haleem, S. Bahl, M. Javaid, S. K. Mahla, and S. Singh, "Automation and manufacturing of smart materials in additive manufacturing technologies using internet of things towards the adoption of industry 4.0," *Materials Today: Proceedings*, 2021.

[13] Y. Wang, Y. Lin, R. Y. Zhong, and X. Xu, "Iot-enabled cloud-based additive manufacturing platform to support rapid product development," *International Journal of Production Research*, vol. 57, no. 12, pp. 3975–3991, 2019.

[14] S. K. Everton, M. Hirsch, P. Stravroulakis, R. K. Leach, and A. T. Clare, "Review of in-situ process monitoring and in-situ metrology for metal additive manufacturing," *Materials & Design*, vol. 95, pp. 431–445, 2016.

[15] S. A. Tofail, E. P. Koumoulos, A. Bandyopadhyay, S. Bose, L. O'Donoghue, and C. Charitidis, "Additive manufacturing: scientific and technological challenges, market uptake and opportunities," *Materials today*, vol. 21, no. 1, pp. 22–37, 2018.

[16] I. Baturynska, O. Semeniuta, and K. Martinsen, "Optimization of process parameters for powder bed fusion additive manufacturing by combination of machine learning and finite element method: A conceptual framework," *Procedia CIRP*, vol. 67, pp. 227–232, 2018.

[17] Z. Zhu, N. Anwer, Q. Huang, and L. Mathieu, "Machine learning in tolerancing for additive manufacturing," *CIRP Annals*, vol. 67, no. 1, pp. 157–160, 2018.

[18] Z. Li, Z. Zhang, J. Shi, and D. Wu, "Prediction of surface roughness in extrusion-based additive manufacturing with machine learning," *Robotics and Computer-Integrated Manufacturing*, vol. 57, pp. 488–495, 2019.

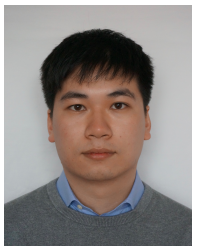
[19] Y. Zhang, J. Y. Fuh, D. Ye, and G. S. Hong, "In-situ monitoring of laser-based pbf via off-axis vision and image processing approaches," *Additive Manufacturing*, vol. 25, pp. 263–274, 2019.
[20] M. Grasso and B. Colosimo, "A statistical learning method for image-based monitoring of the plume signature in laser powder bed fusion," *Robotics and Computer-Integrated Manufacturing*, vol. 57, pp. 103–115, 2019.
[21] J. Alldredge, J. Slotwinski, S. Storck, S. Kim, A. Goldberg, and T. Montalbano, "In-situ monitoring and modeling of metal additive manufacturing powder bed fusion," in *AIP Conference Proceedings*, vol. 1949, no. 1. AIP Publishing LLC, 2018, p. 020007.
[22] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014.
[23] A. Kusiak, "Convolutional and generative adversarial neural networks in manufacturing," *International Journal of Production Research*, vol. 58, no. 5, pp. 1594–1604, 2020.
[24] C. Gobert, E. Arrieta, A. Belmontes, B. R. Wicker, F. Medina, and B. McWilliams, "Conditional generative adversarial networks for in-situ layerwise additive manufacturing data," in *Proceeding of the 29th international Solid Freeform Fabrication Symposium*, 2019.
[25] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.
[26] T. Hinz, M. Fisher, O. Wang, and S. Wermter, "Improved techniques for training single-image gans," *arXiv preprint arXiv:2003.11512*, 2020.
[27] Devicehive. [Online]. Available: https://devicehive.com/
[28] Message queue telemetry transport (mqtt). [Online]. Available: https://mqtt.org
[29] R. T. Fielding and R. N. Taylor, "Principled design of the modern web architecture," *ACM Transactions on Internet Technology (TOIT)*, vol. 2, no. 2, pp. 115–150, 2002.
[30] G. Lawton, "Developing software online with platform-as-a-service technology," *Computer*, vol. 41, no. 6, pp. 13–15, 2008.
[31] W. A. Mustafa and H. Yazid, "Image enhancement technique on contrast variation: a comprehensive review," *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 9, no. 3, pp. 199–204, 2017.
[32] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
[33] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," *arXiv preprint arXiv:1701.04862*, 2017.
[34] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, pp. 6626–6637, 2017.
[35] M. G. Mohammadi, D. Mahmoud, and M. Elbestawi, "On the application of machine learning for defect detection in l-pbf additive manufacturing," *Optics & Laser Technology*, vol. 143, p. 107338, 2021.
[36] T. Meng, X. Jing, Z. Yan, and W. Pedrycz, "A survey on machine learning for data fusion," *Information Fusion*, vol. 57, pp. 115–129, 2020.

Davide Cannizzaro (S'21) is Ph.D. student in Computer and Control Engineering at Politecnico di Torino (Italy). He received a B.Sc Degree in Electronic Engineering and a M.Sc. Degree in Mechatronic Engineering from the same university in 2016 and 2018, respectively. His main research interests concern Artificial Intelligence, Machine Learning and Additive Manufacturing. He is a Student Member of the IEEE.

Antonio Giuseppe Varrella is a Master Student in Artificial Intelligence & Cloud at Politecnico di Torino (Italy). He received the B.Sc in Computer Science at Università degli Studi di Napoli Federico II (Italy) in 2018 and M.Sc in ICT for Smart Societies at Politecnico di Torino (Italy) in 2020. His main research interests are Machine Learning and Big Data.

Stefano Paradiso is Global Additive Manufacturing Development & Networking responsible at Stellantis. He received both B.Sc and M.Sc degrees in Architecture at Politecnico di Torino (Italy) in 2009 and 2012, respectively. His main research interests are Additive Manufacturing and components prototyping in the automotive field.

Roberta Sampieri is the Head of Global Additive Manufacturing Planning & Development at Stellantis. She received M.Sc in Mechanical Engineering at Politecnico di Torino (Italy) in 2003. Her main research interest concerns the Additive Manufacturing innovation and design in the automotive field.



Yukai Chen (M'18) received the M.Sc. degree in computer engineering and the Ph.D. degree in computer and control engineering from the Politecnico di Torino, Turin, Italy, in 2014 and 2018, where he is currently a Postdoctoral Research Fellow. His research interest includes computer-aided design for integrated circuits, cyber physical systems and artificial intelligence.

Alberto Macii (SM'07) received the Laurea and Ph.D. degrees in computer engineering from the Politecnico di Torino (Italy). He is currently a Full Professor of Computer Engineering in the same university. His research interests include design of electronic digital circuits and systems, with a particular emphasis on low-power consumption aspects.

Edoardo Patti (M'16) is Assistant Professor at Politecnico di Torino (Italy). He received both M.Sc. and Ph.D. degrees in Computer Engineering from the same university in 2010 and 2014, respectively. His main research interests concern Ubiquitous Computing, Internet of Things and Smart Systems applications. He is a Member of the IEEE.

Santa Di Cataldo (M'07) is Associate Professor at the Department of Control and Computer Engineering of Politecnico di Torino (Italy). She received a Ph.D. in Computer and Systems Engineering from the same university in 2011. Her main research interests are Artificial Intelligence, Machine Learning and Image Processing. She is a Member of the IEEE.