In-situ quality inspection based on coaxial melt pool images for laser powder bed fusion with depth graph network guided by prior knowledge

Yingjie Zhang [a,*], Honghong Du [a], Kai Zhao [b], Jiali Gao [c], Xiaojun Peng [a], Lang Cheng [a], Canneng Fang [a], Gang Chen [a,*]

[a] *Shien-Ming Wu School of Intelligent Engineering, South China University of Technology, Guangzhou, China*
[b] *Shanghai Aerospace Equipments Manufacture Co., Ltd, Shanghai, China*
[c] *College of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai, China*

ARTICLE INFO

ABSTRACT

In-situ monitoring is crucial for enhancing process quality control in laser powder bed fusion (LPBF). Currently, data-driven approaches in LPBF in-situ quality monitoring have shown remarkable success. However, existing data-driven methods often lack integration with physical knowledge, leading to the opacity of decision-making processes. Research on LPBF knowledge-data mixed-driven modeling is still relatively scarce. To address this gap, this paper proposes a deep graph network method guided by prior knowledge (MK-DGNet) for in-situ quality inspection based on coaxial melt pool images. In the proposed method, prior knowledge is first extracted based on understanding of melt pool. Then, the fusion module is used to place images and knowledge vectors in the same dimensional space. Finally, a deep graph network architecture is elaborately established, taking graph-formatted data as input to learn deep-layer relationships between nodes and edges. The superiority of MK-DGNet is demonstrated using publicly available NIST datasets and self-built CMPQ dataset. Additionally, explainable artificial intelligence methods are employed to explain the basis of network decisions and the effectiveness of prior knowledge. This research provides new methods and perspectives for addressing quality issues in the LPBF process.

1. Introduction

Additive manufacturing (AM), or 3D printing, creates parts directly from computer-aided design (CAD) models by layering materials [1]. This technology enables the production of complex structures that are impossible to achieve with traditional methods. Currently, metal AM is undergoing a paradigm shift from prototyping parts to manufacturing them for final use. Among these technologies, laser powder bed fusion (LPBF) stands out as one of the most promising. LPBF is widely used to manufacture high-precision, high-quality metal products [2,3].

However, LPBF process involves highly complex physical dynamics. Coupling factors such as powder quality, powder thickness, laser power, scanning speed, scan spacing, laser scanning pattern, gas environment, preheating of the build plate, and complexity of the part structure can lead to a high incidence of internal defects and geometric deformations [4–6]. Internal porosity is one of the

* Corresponding authors.

common defects in LPBF-formed parts, significantly compromising part' s mechanical properties[7]. Extensive research has investigated the formation mechanisms and effects of three types of pores: lack-of-fusion, gas pores, and keyholes [8–10]. Hojjatzadeh et al. [11] reported six mechanisms of pore formation in LPBF, including induced pores from small holes, powder material gaps, metal vapor expansion causing boundary pores, surface fluctuations leading to pores, surface shallow depressions forming pores, and crack-induced pores. These findings provide further insights into pore formation in the LPBF process. These internal defects can further lead to decreased part density, degraded mechanical properties, and ultimately reduced repeatability of LPBF process [12]. Therefore, there is an urgent need to develop effective methods for in-situ monitoring of part quality during the printing process.

In-situ quality monitoring aims to prevent potential anomalies in metal additive manufacturing, ensuring that printed parts meet quality standards and avoiding issues such as excessive porosity or low density. It holds immense practical value, allowing for timely detection of internal defects and proactive measures to prevent damage to finished parts. A key method for in-situ quality monitoring in LPBF is "data-driven" approaches. It can leverage statistical and machine learning models to learn from in-situ data and make corresponding output decisions based on factors leading to decreased part quality. Traditional machine learning algorithms using manually extracted features [13–15] and deep learning algorithms utilizing convolutional neural networks [16–18] have achieved promising results, offering possible solutions for reducing porosity, increasing density, and improving part quality. However, if the manually selected features are inappropriate or insufficient, the algorithm's performance may significantly decrease. Although deep neural networks have powerful learning capabilities due to rich network parameters and large-scale data, these networks often lack integration with physical knowledge, resulting in opaque decision-making processes, poor generalization, and limited interpretability.

To improve interpretability, significant effort has been invested in the areas of ex post interpretable analysis and ad hoc interpretable modelling [19]. While post hoc analysis seeks to explain the reasons behind obtained results, ad hoc modeling emphasizes the use of hybrid modeling constructs that combine expertise with neural networks to achieve a transparent decision-making process. Hybrid Driven Modeling(HDM), an ad hoc modeling approach, combines physical information with observational data [20]. HDM operates in three modes: physical information loss function, model architecture design, and input enhancement [21]. Compared to purely data-driven models, research indicates that HDM improves accuracy, generalization, and interpretability [22–24]. In recent years, researchers have also begun to explore the combination of HDM with the LPBF process [25,26]. Lu et al. [27] have developed a feature-based physics-constrained active dictionary learning framework for reconstructing thermal images collected during the fused filament fabrication and melt pool morphology images from LPBF experiments at NIST. This method integrates prior knowledge based on image features, enhancing the quality of reconstructed monitoring signals and improving the performance of state monitoring models. Liu et al. [28] developed an HDM model to predict the porosity by explaining the physical mechanisms and ranges of process parameters affecting porosity formation. Zhu et al. [29] proposed an HDM model that integrates the conservation laws of momentum, mass, and energy with data to adjust the learning process of neural networks, realizing prediction of melt pool characteristics. Du et al. [30] used simulation models for calculating the effects of process parameters and material properties on forming quality, aiming to predict balling phenomena. Ko et al. [31] introduced a physics knowledge-guided machine learning framework to address learning constraints of the process-structure–property (PSP) relationship in AM process. Through two sub-processes: knowledge graph-guided top-down and data-driven bottom-up, the framework is connected to decision and control activities as well as physical and virtual additive systems, achieving systematic PSP knowledge learning. The addition of knowledge provides greater interpretability to the network.

However, most current research transforms knowledge into Euclidean space and combines it with convolutional neural networks (CNNs). Although CNNs have shown effective performance in processing Euclidean spatial data, they have limitations in handling non-Euclidean spatial data. In fact, knowledge or knowledge graphs are more often presented as data in non-Euclidean space [31]. In a knowledge graph, nodes represent entities or concepts, while edges represent relationships or connections between them. Such complex relationships often cannot be simply represented by Euclidean spatial data. A graph is a general data structure where grids and sequences can be seen as special cases [32]. Recently, Graph neural networks (GNNs) have been increasingly applied by researchers to various detection tasks due to the ability to model the interdependencies between graph data [33]. Viewing sensor data as a graph is more flexible and effective for visual perception. Zhang et al. [34] converted acoustic signals into graph and applied a graph convolutional neural network (GCN) to achieve fault diagnosis of rolling bearings. Li et al. [35] transformed vibration signals of rolling bearings into horizontal visibility graph and used GNNs to achieve fault classification. Mozaffar et al. [36] transformed the thermal data obtained from simulation into a graph and used graph recurrent neural networks (GRNN) to capture spatiotemporal dependencies of thermal responses, achieving predictions of thermal histories for different geometries in AM process. Researchers have also attempted to combine GNNs with LPBF. Zhu et al. [37] utilized graph structures to represent relationships between fatigue samples and employed GNNs with a multi-graph attention mechanism to predict high cycle fatigue life of titanium alloys. Jiang et al. [38] used GNNs to predict the tensile strength of lattice structures with different cell topologies. However, the application of GNNs for intelligent quality monitoring in LPBF remains to be explored. Additionally, graphs usually contain prior knowledge influencing the final results [33]. Most works do not consider prior knowledge when constructing graphs. Research on LPBF knowledge-data mixed-driven modeling is still relatively scarce.

Inspired by previous research, this study proposes a deep graph network approach based on melt pool prior knowledge guidance to achieve in-situ quality monitoring in LPBF. It involves fusing sensor data with melt pool prior knowledge and representing the data as a graph of nodes and edges according to the graph structure, allowing the prior knowledge to be involved in network training. However, testing the effectiveness of the prior knowledge poses a challenge. To address this limitation, we employ explainable artificial intelligence methods to evaluate the validity of the knowledge. The proposed depth graph network approach aims to incorporate the prior knowledge and a more generalized form of data in-situ monitoring printing quality. The main innovations and contributions of the methodology can be summarized as follows:

(1) We designed a deep graph neural network to achieve coaxial melt pool quality monitoring for LPBF. The proposed model was validated using the publicly available NIST dataset and the internally created CMPQ dataset. By processing data through a more flexible and generalized graph structure approach, it offers an alternative to traditional deep convolutional networks. This work provides a potential solution for real-time monitoring in LPBF.

(2) We designed a approach incorporating prior knowledge with raw sensor data. We selected physical, regional, and statistical information affecting melt pool quality as sources of prior knowledge and represented the data as a graph structure of nodes and edges, which participated in network training. The overall classification accuracy was improved, with significant enhancements observed in certain categories.

(3) We employed two explainable artificial intelligence methods to elucidate the basis of network decisions and the effectiveness and importance of incorporated prior knowledge. Visualizing the feature maps explained the quality of network feature extraction. By visualizing the attribution scores of melt pool images and prior knowledge, we explained the relationship between melt pool morphology and quality, enhanced our understanding of the model's decision-making process.

The remaining sections of the paper are structured as follows. Section 2 provides detailed information on the NIST and CMPQ datasets construction. Section 3 outlines the structure of the proposed methodology. Section 4 presents a comprehensive analysis of the performance and experimental results. Finally, Section 5 presents the conclusions and future work.

2. Datasets construction

The models presented in this paper were trained and validated using two datasets: the NIST dataset, as described in detail in [39], and the CMPQ dataset developed by the authors of this publication. Table 1 summarizes the constant parameters used during the creation of both datasets. The melt pool images of both datasets were acquired by the coaxial monitoring system, and the schematic diagram of the system is shown in Fig. 1. Coaxial melt pool images were captured by a modified high-speed camera mounted coaxially with the laser beam path. It means melt pool appears stationary within the field of view. The coaxial melt pool image acquisition system monitors melt pool by capturing morphology image frames $f_i \in R^{W \times H}$, $i = 1, 2, \cdots, N$ with a near-infrared high-speed camera. W and H are the width and height of the image and N is the total number of melt pool images.

2.1. NIST dataset

The NIST dataset contains 12 cubes with dimensions of 10 mm × 10 mm × 5 mm. Each cube is printed with different scanning strategies and process parameters and is accompanied by a corresponding XYPT command file, including the x and y positions of the laser, power and trigger, as shown in Fig. 2. The coaxial imaging system for the NIST dataset is synchronized with the controller of the AM machine, capturing melt pool image frames when the trigger is set to 2. Comparison of the stored melt pool images with the markers of the XYPT command file shows that the image sequence is highly consistent with the triggers in the XYPT file. This alignment ensures the reliability and consistency of the acquired data. The XYPT file can be used to determine the location of the acquired melt pool image and the laser power at that point in time.

Furthermore, to ascertain the scanning speed associated with the melt pool image denoted as F_i, where i represents the frame number, it can be obtained through the laser positions S in frames i and i-1, and the interval time T between the two frames using the following equation:

$$v_i = \frac{\|S_i - S_{i-1}\|}{T} \tag{1}$$

The laser positions can be directly extracted from the XYPT file, while T can be calculated by multiplying the recording interval of the triggers in the XYPT file by 10 us, where 10 us represents the time step for recording in the XYPT file [39].

Therefore, each melt pool image can be associated with its corresponding laser power and scanning speed. Ultimately, The NIST melt pool image dataset in this paper was established comprising 7 different combinations of laser power levels and scanning speeds in: P100_V900, P156_V800, P166_V500, P195_V500, P195_V800, P234_V800, and P60_V25, where 'P' denotes laser power (W) and 'V' denotes scanning speed (mm/s). The original data included over 40,000 combinations of different power and speed. Based on the number of image samples in each category and the actual scanning strategy, the 7 specific parameter combinations listed above were chosen as classification labels. However, there is still a significant difference in sample sizes among the selected 7 categories. For instance, the category P195_V800 with nominal laser power and scanning speed has over 240,000 samples, whereas the category P60_V25 has only about 1,000 samples, making the former approximately 240 times more numerous than the latter. To facilitate a more intuitive comparison of model performance and account for the impact of sample size on model training time, we randomly

Table 1
Constant parameters for NIST and CMPQ datasets.

	Substrate material	Powder material	Layer thickness	Laser beam diameter	Bandwidth filter	Pixel size	Frame rate	Integration Time
NIST	IN625	IN625	20 μm	85 μm	830–870 nm	8 μm	10KHZ	20 μs
CMPQ	316L	316L	50 μm	70 μm	700–950 nm	4.3 μm	3KHZ	0.25 ms

Fig. 1. Coaxial melt pool monitoring system.

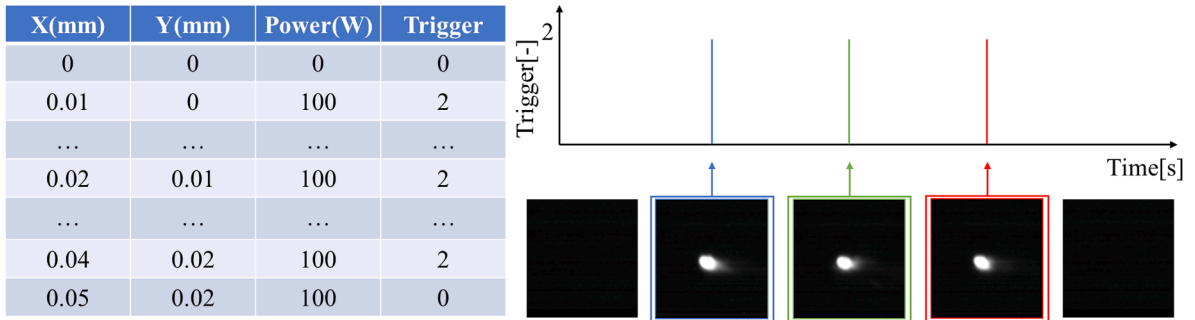| X(mm)	Y(mm)	Power(W)	Trigger
0	0	0	0
0.01	0	100	2
…	…	…	…
0.02	0.01	100	2
…	…	…	…
0.04	0.02	100	2
0.05	0.02	100	0

Fig. 2. XYPT file and corresponding melt pool image acquisition.
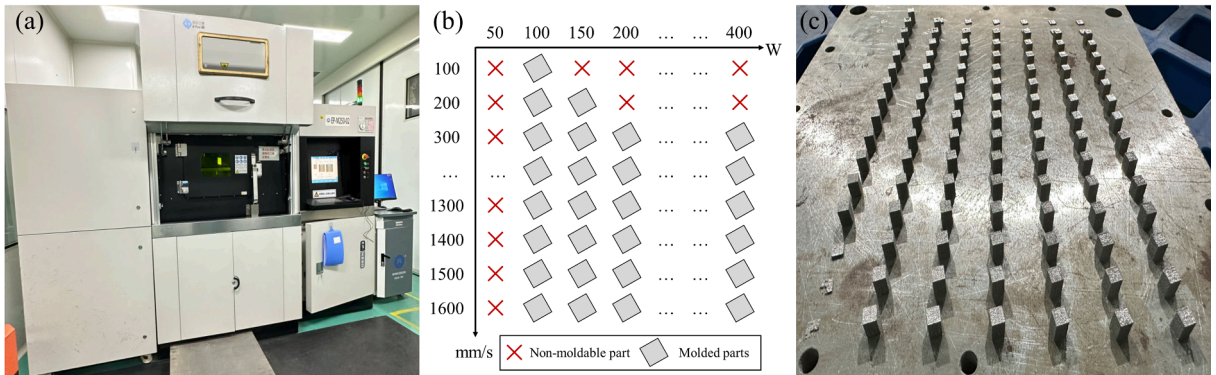

Fig. 3. (a) LPBF equipment. (b) Experimental parameter design. (c) Experimental final molding results.

selected 900 melt pool images representing different scanning strategies for each category, ultimately forming a 7-class balanced dataset with 6,300 samples. The dataset was then split into training and validation sets in 9:1 ratio, with 5,670 melt pool images in the training set and 630 images in the validation set. It is worth noting that this remains a challenging classification task. Due to variations in scanning path and thermal conditions, even with the same laser power and scanning speed, there is significant uncertainty in the shape and size of the melt pool, with the melt pool area varying up to five times [40]. In the constructed NIST dataset, each category includes the effects of different scanning strategies, there is still considerable uncertainty in melt pool images.

2.2. CMPQ dataset

2.2.1. Experiment setup and data registration

The experiments were carried out on an EP-M250 (Hangzhou Shining Eplus 3D Tech Co., Ltd.), and the appearance of the machine is shown in Fig. 3(a). The build size of the print chamber is $262 \times 250 \times 300 \text{ mm}^3$. We printed $5 \text{ mm} \times 5 \text{ mm} \times 5 \text{ mm}$ cube parts using 128 combinations of laser power (ranging from 50 to 400 W) and scanning speed (ranging from 100 to 1600 mm/s) to comprehensively cover the different energy density, producing parts with varying qualities. The energy density is defined as follows:

$$E = \frac{P}{V.H.T} \tag{2}$$

where P represents laser power, V represents scanning speed, H represents hatching distance, and T represents layer thickness. The specific parameters are shown in Fig. 3(b). The raw material was 316L stainless steel powder with a particle size of about 5–53 μm. Table 2 summarizes the actual chemical composition content as well as the particle size distribution of the 316L powders in the experiments. The layer thickness was set to 50 μm, the layer rotation angle to 90°, and the hatching distance to 110 μm. In addition, insufficient energy density leads to lack of fusion and incomplete parts. Excessive energy density can cause severe super-elevation and over melt, disrupting the formation of other parts in subsequent layers. It's necessary to remove parts that cannot be formed or experience severe super-elevation during the printing process. The experiment stabilized after printing 20 layers, and there was no need to unload additional parts afterward. The final results, shown in Fig. 3(c), included 98 successfully printed parts.

In self-built coaxial melt pool monitoring system, the high-speed camera is the Ximea xiQ–USB3 Vision Camera, MQ013xG-ON model, with the resolution of 96×100 pixels and the pixel size of 4.8 μm. The frame rate is 3000 fps, and the exposure time is set to 0.25 ms. This acquisition system samples using its own clock and is not synchronized with the AM machine's controller. In general, LPBF system determines the geometry and location of the part based on the imported design file. Recognizing part features, the system slices the part into multiple layers and generates the actual scan path for each layer based on the contours of each slice and the user's scanning strategy. In this study, serpentine scanning strategy is used for scanning parts. Laser shuts off briefly and melt pool disappears when laser moves beyond the part's contour during the printing process. This duration is defined as the turning time t_{turn}. During this period, the gray level of melt pool image captured is much lower than when laser is turned on. A similar pattern is observed when the laser moves to the starting point of scanning the next part after completing the previous part, defined as the transition time $t_{transition}$. Transition time is much greater than turning time, i.e., $t_{transition} \gg t_{turn}$. Therefore, t_{turn} and $t_{transition}$ are used to differentiate melt pool images into different sets corresponding to different paths and parts as shown in Fig. 4.

2.2.2. CMPQ dataset construction

In real-world scenarios, we usually focus more on the quality of the printed parts. Therefore, in-situ monitoring data have frequently been used to detect defects, predict or control AM part quality [41,42]. According to the type of porosity and density, parts are generally classified into different labels such as high quality, medium quality, low quality or lack of fusion, conductive modes and keyholes. This type of research focuses on linking sensor data generated by combinations of process parameters to part quality, which helps identify changes in part quality caused by slight perturbations in process parameters. To facilitate comparative analysis with existing LPBF research, we also mapped in-situ monitoring data to part quality in this study, using post-processed quality labels of high, medium, and low. However, merely using high, medium, and low labels might lead the model to find shortcut solutions, resulting in simplification. Therefore, in the ablation experiment, we also conducted a study on directly identifying subtle changes in process parameters similar to the labels used in NIST dataset for ensuring that our results are not affected by model oversimplification.

Currently, the industry standard for monitoring the quality of parts related to porosity and density is X-ray. However, obtaining density and porosity data for multiple samples via X-ray is both expensive and time-consuming. The Archimedes method provides a faster and more economical way to collect density data for multiple samples. To reduce measurement errors, the experiment was repeated five times for each sample, ensuring the sample remained dry and clean throughout. Obvious outliers due to improper operation were excluded, ensuring at least three valid measurements per sample. The sample's density (ρ) and relative density (η) is defined as follows:

Table 2
Chemical compositions and particle size distribution of 316L powder.

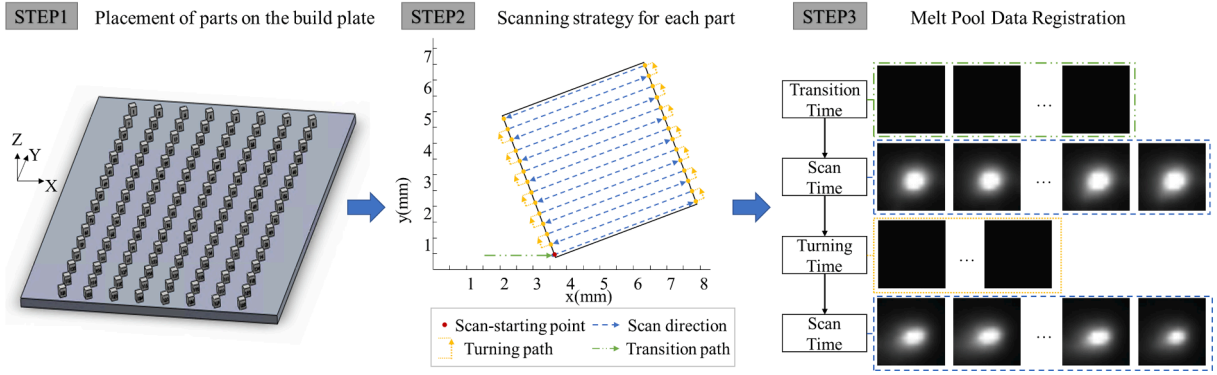| Chemical Composition (%)									Particle Size Distribution (μm)		
Si	C	Cr	S	Ni	P	Mo	Mn	Fe	D10	D50	D90
0.38	0.007	17.08	0.003	10.73	0.021	2.42	0.45	Bal.	19.10	33.10	56.50

Fig. 4. Laser scanning path and Melt Pool data registration.

$$\rho = \frac{m_1}{m_1 - m_2} \times \rho_{liquid}, \eta = \frac{\rho}{\rho'} \tag{3}$$

where m_1 represents the mass of the sample in air (g), m_2 represents the mass of the sample is immersed in liquid (g), ρ_{liquid} represents the density of liquids (g/cm^3), and ρ' represents the theoretical density (g/cm^3). In this experiment, the liquid was distilled water $(\rho_{liquid} = 1\text{g/cm}^3)$. The theoretical density of 316L stainless steel is 7.98g/cm^3.

Fig. 5(a) illustrates the histogram of part's relative density. Based on these measurements, parts were annotated and paired with corresponding melt pool images to create a labeled dataset, termed the Coaxial Melt Pool Quality (CMPQ) dataset. This dataset categorizes parts into three quality classes: high quality (density $\geq$98 %), medium quality (95 % $\leq$density <98 %), and low quality (density $\leq$95 %). The thresholds of 95 % and 98 % were determined by considering the average distribution of image data and actual working conditions. As shown in Fig. 5(b), the distribution of samples across different quality classes in the CMPQ dataset is depicted. From each part, 200 images were randomly selected as corresponding melt pool data, resulting in 19,600 melt pool images in total. The dataset was then split into training and validation sets in a 9:1 ratio, with 17,640 melt pool images in the training set and 1,960 images in the validation set.

3. Methodology

The method aims to achieve in-situ quality inspection based on deep graph networks guided by melt pool prior knowledge (MK-DGNet). Fig. 6 illustrates the general framework of the proposed method, which consists of two main components: knowledge vector acquisition and depth graph network construction. By combining prior knowledge with image data and transforming it into graph representation, its advantages include: 1) Transparent decision-making process achieved by combining domain expert knowledge with neural networks; 2) Decision-making using graph structures allows for more flexible modeling of complex objects, as graph is a generalized data structure where grids and sequences can be viewed as special cases; 3) Advanced research in Graph Neural Networks (GNNs) can be transferred to online monitoring in LPBF.
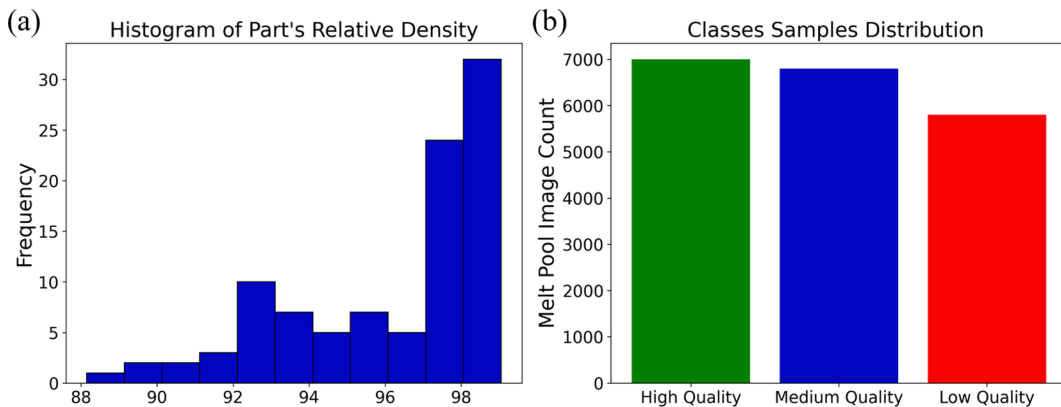

Fig. 5. (a) Histogram of part relative density. (b) Distribution of quality classes in CMPQ dataset.
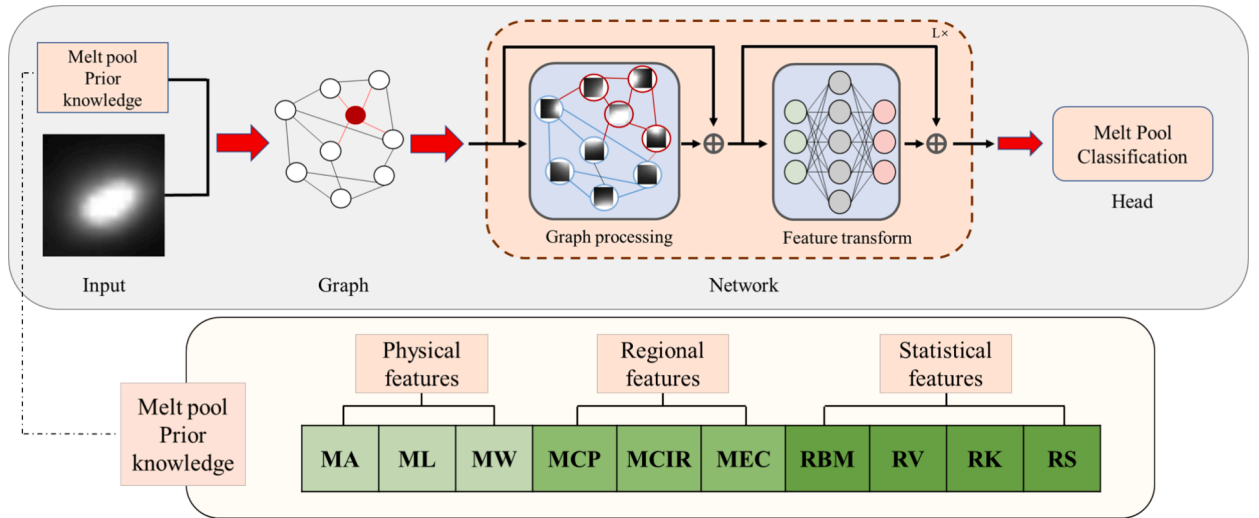
Fig. 6. The framework of the proposed MK-DGNet.

3.1. Melt pool prior knowledge

The interpretation of "knowledge" varies across different domains. In linguistic, knowledge is understood as the comprehension or information about a subject acquired through experience or research. In additive manufacturing, "knowledge" pertains to a repository of process models, statistical inferences, data, or other information sources that can be extrapolated and transferred across materials, machines, and part types. Melt pool refers to the liquid area formed when a laser irradiates the surface of metal powder. The quality and stability of melt pool directly affect the microstructure and mechanical properties of the final part. Monitoring the melt pool in LPBF and other metal 3D printing processes is crucial, as various melt pool information has been widely utilized in online quality inspection of LPBF processes [43–45]. In this study, melt pool serves as the object for acquiring prior knowledge, represented as a one-dimensional vector. Three physically intuitive melt pool features, three region features, and four statistically based general features are designed, representing the melt pool prior knowledge to guide network training. It is worth noting that the form of prior knowledge is not entirely fixed, and with deeper understanding of the LPBF process, higher-dimensional knowledge can be extracted for subsequent integration into network training to guide the training process further.

Specifically, as depicted in Fig. 7, these features were obtained through the extraction of the region of interest (ROI), median filtering, and binary thresholding. The center of the melt pool corresponds to the point of direct contact between the laser and the material, where the temperature is highest, energy density is maximum, and pixel brightness is highest. By summing each pixel along rows and columns separately, the maximum pixel values in the horizontal and vertical directions are located to determine the position of melt pool center. Using this center as reference, 32×32 size region of interest (ROI) is cropped to ensure extraction of the primary area containing melt pool. The 32×32 ROI size was determined through repeated experiments based on the characteristics of the NIST
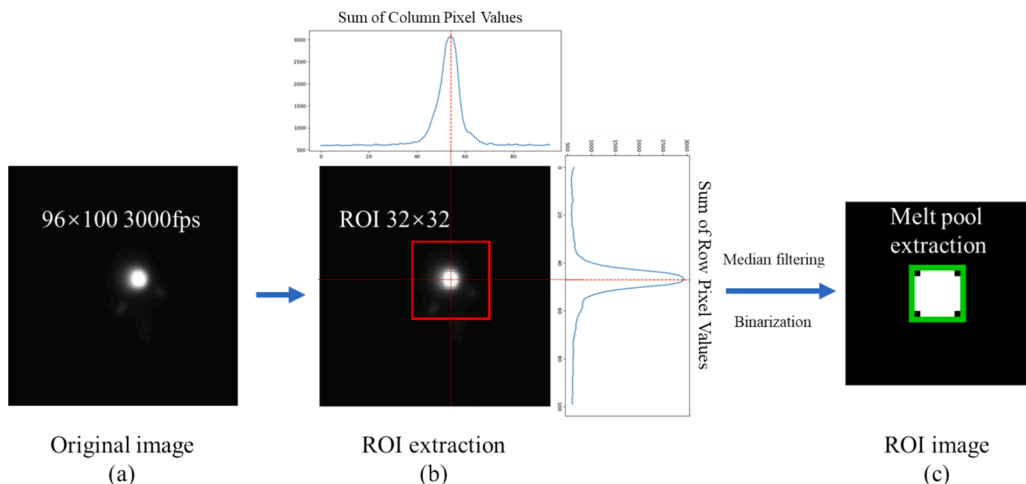

Fig. 7. Process of melt pool feature extraction.

and CMPQ datasets. Additionally, median filtering with a 3×3 filter kernel size reduces image noise and enhances melt pool effect. During binarization, an adaptive thresholding method is utilized to fully extract the melt pool region and generate its bounding rectangle. Subsequently, the area, length, and width of the melt pool are calculated. Melt pool area (*MA*) can be computed using Green's formula:

$$MA = \oint_C xdy - ydx \tag{4}$$

where C represents the contour of melt pool, and x, y are coordinate functions of the contour curve C. Additionally, melt pool length (*ML*) and melt pool width (*MW*) are determined by identifying the minimum area bounding rectangle enclosing melt pool region. *ML* denotes the length of this rectangular boundary, and *ML* denotes its width.

Region features summarize the key geometric characteristics, aiding in deeper understanding of structural properties. Compactness, circularity, and eccentricity are crucial for learning melt pool characteristics. Melt pool compactness (*MCP*) measures the degree of pixel aggregation within melt pool, where higher compactness indicates clear boundaries and regular shapes. *MCP* is defined as:

$$MCP = \frac{Perimeter^2}{MA} \tag{5}$$

where Perimeter denotes melt pool perimeter. On the other hand, melt pool circularity is crucial for evaluating shape consistency. Melt pool with lower circularity values may exhibit irregular shapes or deviate from the ideal state, potentially impacting part's quality. Melt pool circularity (*MCIR*) holds particular significance in detecting irregular and abnormal melt pool. *MCIR* is calculated by the following formula:

$$MCIR = \frac{4\pi \bullet MA}{Perimeter^2} \tag{6}$$

Although MCP and MCIR have similar expressions, they have different region meanings. The network can capture their deep relationships through complex feature extraction. Eccentricity is also one of the key features describing the shape of melt pool. Lower eccentricity values indicate that the melt pool shape is close to circular, while higher eccentricity values indicate that the melt pool has a more elongated or stretched shape. Melt pool eccentricity (*MEC*) can be defined as the ratio of the ellipse's focal length (*2c*) to its major axis (*2a*), which is represented as follows:

$$MEC = \frac{2c}{2a} = \frac{\sqrt{a^2 - b^2}}{a} = \sqrt{1 - \left(\frac{b}{a}\right)^2} \tag{7}$$

where *b* represents the length of the semi-minor axis. Additionally, statistical features comprehensively summarize the distribution characteristics of melt pool image intensity, providing valuable prior information for subsequent analysis and decision-making. These features include ROI's brightness mean (*RBM*), variance (*RV*), kurtosis (*RK*), and skewness (*RS*). *RBM* and *RV* can reflect the brightness and uniformity of the melt pool, while *RK* and *RS* can unveil the shape and symmetry of the intensity distribution of the melt pool.
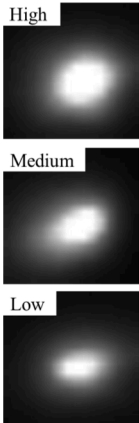
Prior Knowledge vector		Physical features			Regional features			Statistical features			
		MA	ML	MW	MCP	MCIR	MEC	RBM	RV	RK	RS
High	Original vector	36.5	8	8	15.65	0.80	0.56	46.14	3483.36	4.94	2.38
	Normalized vector	.1740	.0909	.2500	.0005	.8509	.5625	.7651	.7115	.2157	.4270
Medium	Original vector	24	7	6	16.10	0.78	0.68	39.27	2624.02	7.44	2.76
	Normalized vector	.1135	.0727	.1667	.0006	.8270	.6830	.6148	.5357	.3124	.5251
Low	Original vector	15	7	4	18.88	0.67	0.84	34.68	1873.67	10.35	3.11
	Normalized vector	.0701	.0727	.0834	.0013	.7047	.8367	.5142	.3823	.4250	.6152

Fig. 8. Knowledge vectors for different melt pool quality.

The comprehensive analysis of intuitive physical features, region features, and statistical features enables a thorough assessment of the morphology, structure, and quality of the melt pool, providing researchers with crucial reference information. This aids in optimizing the melt pool formation process and improving the quality of parts. In this study, the aforementioned 10 parameters were combined into a one-dimensional prior knowledge vector. To ensure fair comparison, input data were normalized to the range [0, 1] using min–max normalization:

$$\mathrm{x}' = \frac{x - min(x)}{max(x) - min(x)} \tag{8}$$

This normalization scales the numerical range of each parameter, avoiding biases introduced by different magnitudes and ensuring equal treatment during model training. Fig. 8 illustrates the knowledge vectors of melt pools with different qualities.

3.2. Fusion of prior knowledge and images

The depth graph network designed in this study utilizes a dual-input framework of images and knowledge vectors. In this framework, knowledge vectors are represented as one-dimensional vectors of length 10, while images are two-dimensional information. To effectively integrate prior knowledge into the network training process, we introduce fusion module. This module merges one-dimensional and two-dimensional information, aligning both the image and the knowledge vector within the same dimensional space.

The inspiration for this design stems from the application of fully connected (FC) layer neurons in deep learning, where one-dimensional information is typically processed. Typically, 2–3 FC modules are employed for classification tasks, with the number of neurons in the final FC layer being fewer than the preceding 1–2 layers. The rationale behind this design is to gradually reduce the complexity. In this study, we reverse this concept. Firstly, we expand one-dimensional vectors to increase their node count, and then transform them into two-dimensional pseudo-images. Meanwhile, by leveraging the downsampling and upsampling capabilities of convolutional and deconvolutional layers, we can flexibly adjust their sizes and channel numbers to scale the original images and pseudo-images to a fixed size. The purpose of this process is to uniformly process data from different input sources and enable seamless interaction within the network.

Specifically, the Vector-Transform operation first applies two linear transformations to the input vector of length 10, mapping it to lengths of 24, and 49. Subsequently, the vector of length 49 is adjusted to form a pseudo-image of size 7 × 7. The pseudo-image undergoes two transposed convolutions for upsampling, resulting in final size of 56 × 56 and the channel count of 16. Choosing a smaller channel count aims to reduce the model's parameter count. During the image transformation process, the input image size was first adjusted to 224 × 224 to align with the conventional input size used by classic end-to-end CNNs. If the size is not adjusted before input, the structure of CNNs would require significant modification to function properly, which could impact their performance. This adjustment facilitates subsequent comparative analysis. Then, we applied suitable convolution operation to deflate the image size, ultimately obtaining an output with dimensions of 64 × 56 × 56. The sizes of knowledge vectors and melt pool image changes in the fusion module are shown in Fig. 9. Conv-K-S-P-C denotes that the size of convolution kernel is K × K, the size of stride is S, the size of padding is P, and the number of channels is C. Up-Conv-K-S-P-C denotes transposition convolution, and the meaning of the parameters is the same as that of convolution kernel.

3.3. Depth graph network construction

For data of size H × W × C, we could divide it into N data blocks [46]. Each data block is transformed into a feature vector $x_i \in R^D$,
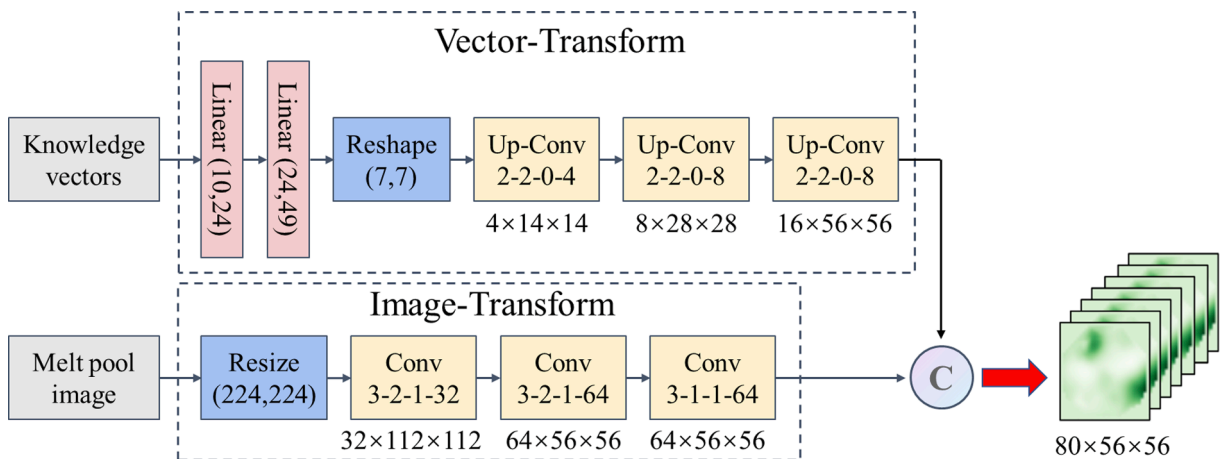

Fig. 9. Fusion module setting.

resulting in R $= [x_1 \quad x_2 \cdots \quad x_N]$, where D is the dimension of the features, and $i = 1, 2, \cdots, N$. These features can be considered as a set of unordered nodes. The number of nodes N is set as 196. The distance $d(x_i, x_j)$ between nodes $x_i, x_j \in R[x_N]$ is computed as follows:

$$d(x_i, x_j) = \|x_i - x_j\|^2 = \sum_{k=1}^{D} (x_{ik} - x_{jk})^2 \tag{9}$$

By comparing the distance between x_i and other nodes, We can find K nearest neighbors $N(x_i)$ for x_i. For each $x_j \in N(x_i)$, we create an edge e_{ji} from x_j to x_i, where the edge weight is the distance between the two nodes. Then we obtain a graph G $= (X, E)$, where E represents all the edges.

For a graph G $= (X, E)$, graph convolutional layers can leverage the topology and connectivity of nodes to extract high-dimensional features from the graph. The core of the graph convolution operation is to aggregate the features of each node's neighboring nodes [32,47]. Through this process, each node integrates information from its surrounding nodes, reflecting it in its own feature representation. This aggregation and updating process aids in capturing both the global structure and local features of the graph, providing rich input features for subsequent graph-related tasks. Specifically, graph convolution operates as follows:

$$G' = F(G, W) = Update(Aggregate(G, W_{agg}), W_{update}) \tag{10}$$

where W_{agg} and W_{update} are the learnable weights of the aggregation and update operations, respectively. More specifically, the aggregation operation computes a representation of a node by aggregating features of neighboring nodes, and the update operation further merges the aggregated features:

$$x_i' = h(x_i, g(x_i, \mathcal{N}(x_i), W_{agg}), W_{update}) \tag{11}$$

where $\mathcal{N}(x_i^l)$ denotes the set of neighboring nodes of x_i^l. To demonstrate the flexibility and effectiveness of the method, we adopt the maximum relative graph convolution as the operation for graph convolution, as it is both simple and efficient [48], which is represented as follows:

$$g(\bullet) = x_i'' = [x_i, max(\{x_j - x_i | j \in \mathcal{N}(x_i)\})] \tag{12}$$

$$h(\bullet) = x_i' = x_i'' W_{update} \tag{13}$$

The bias term is omitted. The graph-level processing described above can be represented as:

$$X' = GraphConv(X) \tag{14}$$

Further, the multi-head update operation of graph convolution is introduced. The aggregated feature x_i' is first divided into h heads, namely $head^1, head^2, \ldots, head^h$. Then, these heads are updated separately using different weights. All heads can be updated in parallel and concatenated to form the final value.

$$X' = \left[head^1 W_{update}^1, head^2 W_{update}^2, \cdots, head^h W_{update}^h \right] \tag{15}$$

The multi-head update operation allows the model to update information in multiple representation subspaces, promoting feature diversity. Also, linear layers can be applied before and after graph convolution to project node features into the same domain and increase feature diversity. Additionally, Non-linear activation functions is inserted after graph convolution to prevent layer collapse. We refer to the upgraded module as the "Grapher" module. In practice, given input features $X \in R^{N \times D}$, the Grapher module can be represented as:

$$Y = \sigma(GraphConv(XW_{in})W_{out} + X \tag{16}$$

where $Y \in R^{N \times D}$, W_{in} and W_{out} are the weights of the fully connected layers, and σ is the activation function. To further enhance the feature transformation capability and alleviate the over-smoothing phenomenon, we employ a feed-forward network (FFN) at each node. The FFN module is a simple multi-layer perceptron consisting of two fully connected layers:

$$Z = \sigma(YW_1)W_2 + Y \tag{17}$$

where $Y \in R^{N \times D}$, W_1 and W_2 are the weights of the fully connected layers, and the bias terms are omitted. The hidden dimension of the FFN is typically greater than D. In both the Grapher and FFN modules, batch normalization is applied after each fully connected layer or graph convolutional layer. The stacking of Grapher modules and FFN modules forms the basic building blocks of the Depth-Graph-Net block.

Based on the data's graph representation and the proposed Depth-Graph-Net block, we can construct the deep graph network guided by melt pool prior knowledge (MK-DGNet) for the task of in-situ quality inspection. Referring to the pyramid structure of traditional deep convolutional networks, as the network deepens, the spatial dimensions of the feature maps gradually decrease, which leverages the scale invariance of images to generate high-dimensional features. Specifically, as the network progresses, the input data dimensions (H, W, C) will see reductions in H and W while the dimension C increases. The data size variations of MK-DGNet are summarized in Table 3.

4. Result and discussion

In this study, we evaluated 10 different deep learning models, including models that solely used melt pool images for classification: VGG16 [49], GoogLeNet [50], ResNet50 [51], MobileNetV2 [52], and DGNet, as well as their improved versions. The enhanced versions involved integrating melt pool prior knowledge into the aforementioned models and incorporating it into network training. These improved models are respectively named MK-VNet, MK-GNet, MK-RNet, MK-MMet, and the proposed MK-DGNet. VGG16 extracts features by increasing the depth of the network, with the classification part employing three fully connected layers for classification. GoogLeNet extracts information from images at different scales through multiple convolutional kernels, and finally merges them to obtain different representations of the image. ResNet50 utilizes residual blocks to extract features, and the classification part consists of a combination of Global Average Pooling (GAP) and one fully connected layer. MobileNetV2 is characterized by faster training speed and fewer parameters, making it suitable for deployment on mobile devices. MK-DGNet and DGNet capture deep features of the data in graph manner, with MK-DGNet integrating melt pool prior knowledge to guide network training.

We evaluated the performance of our models using both the publicly available NIST dataset and the dataset we created ourselves, named CMPQ. To ensure fair comparison, they were trained on a single NVIDIA RTX 3060 GPU and all models were trained for 100 epochs with a batch size of 32. We utilized the Adam optimizer with a learning rate set to 0.001 for all experiments. Additionally, we incorporated label smoothing techniques into the model training process. This technique involves smoothing the cross-entropy loss between predicted probabilities and ground truth labels, encouraging the model to produce more calibrated probabilities and reducing the risk of overfitting to noisy labels. The loss function is represented as follows:

$$Loss = -\frac{1}{N}\sum_{i=1}^{N}(1-\alpha) \bullet y_i \bullet log(\widehat{y_i}) - \alpha \bullet \frac{1}{N}\sum_{i=1}^{N}log(\widehat{y_i}) \qquad (18)$$

where y represents the probability distribution of the ground truth labels, y_i is the probability of the i-th class, $\widehat{y_i}$ signifies the model's predicted probability for the i-th class, and α is the smoothing factor set to 0.01.

4.1. Model performance

Table 4 presents the accuracy (ACC) and F1-score (F1) of different models on the NIST and CMPQ datasets, as well as the states of the models (underfitting UF, overfitting OF, normal fitting NF, oscillation OSC). The proposed MK-DGNet performs excellently on the NIST dataset, achieving the highest ACC and F1 among all the compared models, with values of 85.56 % and 85.60 %, respectively. Compared to traditional CNNs, MK-DGNet shows significant performance improvements in ACC ranging from 5.09 % to 12.54 % and in F1 ranging from 4.95 % to 12.43 %. The ACCs of traditional CNNs (VGG16, GoogLeNet, ResNet50, and MobileNetV2) are 80.47 %, 73.02 %, 77.78 %, and 76.03 %, respectively, and their F1 are 80.65 %, 73.17 %, 77.80 %, and 75.94 %, respectively. Compared to the large-parameter VGG16, MK-DGNet's ACC and F1 improved by 5.09 % and 4.95 %, respectively, while reducing the parameter count by 108.36 MB and the number of FLOPs by 11.22G. Compared to ResNet50 with similar number of parameters and FLOPs, MK-DGNet's ACC and F1 improved by 7.78 % and 7.8 %. Compared to GoogLeNet and the lightweight MobileNetV2, MK-DGNet's ACC improved by 12.54 % and 9.53 %, and F1 improved by 12.43 % and 9.66 %, respectively. On the self-created CMPQ dataset, compared to traditional CNNs, MK-DGNet also maintained the highest ACC and F1 at 96.34 % and 96.37 %, showing improvements ranged from 1.52 % to 4 %, and ranged from 1.53 % to 3.91 %, respectively. Compared to VGG16's 93.83 % and 93.87 %, MK-DGNet's ACC and F1 improved by 2.51 % and 2.5 %; compared to GoogLeNet's 92.95 % and 93.05 %, they improved by 3.39 % and 3.32 %; compared to ResNet50's 94.85 % and 94.84 %, they improved by 1.5 % and 1.53 %; compared to MobileNetV2's 92.34 % and 92.46 %, they improved by 4 %

Table 3

Size variation of MK-DGNet. H × W: input data size, D: feature dimension, E: hidden dimension ratio in FFN, K: number of neighbors in graph convolution.

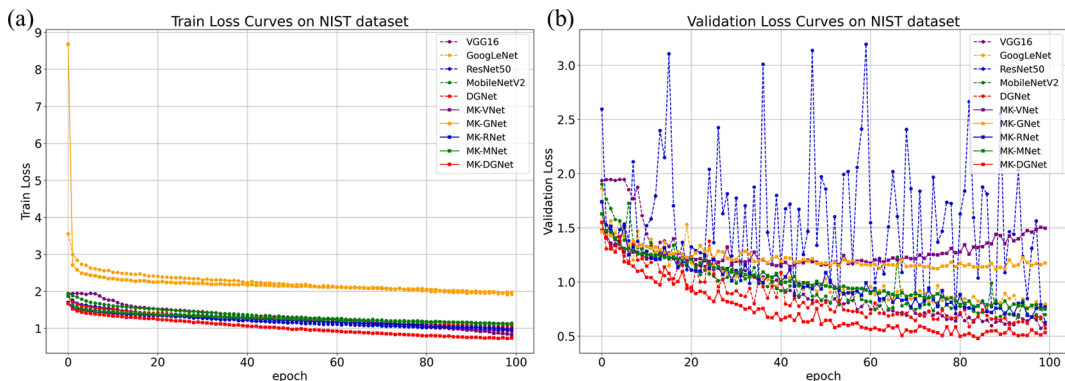| Stage	Output size	MK-DGNet
Fusion module	H/4 × W/4	Vector-Transform Image-Transform
Stage1	H/4 × W/4	$\begin{bmatrix} D=80 \\ E=4 \\ K=9 \end{bmatrix} \times 2$
Downsample	H/8 × W/8	Conv-3–2–1–160
Stage2	H/8 × W/8	$\begin{bmatrix} D=160 \\ E=4 \\ K=9 \end{bmatrix} \times 2$
Downsample	H/16 × W/16	Conv-3–2–1–160
Stage3	H/16 × W/16	$\begin{bmatrix} D=160 \\ E=4 \\ K=9 \end{bmatrix} \times 6$
Downsample	H/32 × W/32	Conv-3–2–1–640
Stage4	H/32 × W/32	$\begin{bmatrix} D=640 \\ E=4 \\ K=9 \end{bmatrix} \times 2$
Head	1 × 1	Pooling&MLP

Table 4

Classification performance of different model on NIST, CMPQ dataset.

Model	FLOPs (G)	Parameter (M)	NIST			CMPQ			Validation Time (s/batch)
			ACC (%)	F1 (%)	Model State	ACC (%)	F1 (%)	Model State	
VGG16	15.47	134.29	80.47	80.65	NF	93.83	93.87	NF	0.146
GoogLeNet	1.58	5.98	73.02	73.17	NF	92.95	93.05	NF	0.140
ResNet50	4.13	23.52	77.78	77.80	OSC	94.85	94.84	OSC	0.183
MobileNetV2	0.33	2.23	76.03	75.94	NF	92.34	92.46	NF	0.122
DGNet	4.36	25.98	79.52	79.43	NF	94.13	94.20	NF	0.135
MK-VNet	15.24	134.27	53.49	52.73	OF	92.45	92.53	NF	0.219
MK-GNet	1.55	5.98	54.76	53.12	UF	86.17	86.28	NF	0.126
MK-RNet	4.10	23.52	76.67	76.88	NF	95.50	95.49	NF	0.192
MK-MNet	0.35	2.26	73.49	73.52	NF	93.21	93.29	NF	0.141
MK-DGNet	4.25	25.93	**85.56**	**85.60**	NF	**96.34**	**96.37**	NF	0.166

and 3.91 %. MK-DGNet consistently achieved the highest ACC and F1 on both datasets. Considering the overall performance in terms of parameter size, computational load, and model performance, MK-DGNet achieves optimal results with relatively lower computational and storage consumption, effectively reducing memory usage and saving computational resources. It is worth noting that while the runtime in the table does not show significant differences during the testing of each model, actual deployment time can be influenced by various factors such as GPU computation speed and data loading methods.

To further illustrate the effectiveness of the proposed MK-DGNet, Figs. 10 and 11 show the convergence of loss with epochs for different models on the NIST and CMPQ datasets. Compared to other models, MK-DGNet achieves the lowest training and validation loss on both the NIST and CMPQ datasets. Specifically, on the NIST dataset, compared to other models, MK-DGNet maintains the lowest training and validation loss levels after the 20th epoch. On the CMPQ dataset, MK-DGNet's training and validation losses nearly remained at the lowest levels after the 20th epoch. MK-DGNet not only performs excellently on both datasets but also demonstrates effective optimization during training. Fig. 12 show the changes of validation accuracy with epochs for different models on the NIST and CMPQ datasets. Compared to other models, MK-DGNet's validation accuracy increases rapidly and remains at the highest level after the 20th epoch, consistent with the changes in loss.

Furthermore, comparing CNNs that incorporate prior knowledge of melt pool (MK-VNet, MK-GNet, MK-RNet, MK-MMet) with corresponding traditional CNNs (VGG16, GoogLeNet, ResNet50, MobileNetV2), we observed significant differences in performance. Specifically, MK-VNet and MK-GNet performed poorly compared to VGG16 and GoogLeNet on both datasets. On the NIST dataset, MK-VNet's validation loss increased with more epochs, indicating clear overfitting. Meanwhile, MK-GNet's loss did not decrease significantly, and its ACC and F1 were relatively low, indicating underfitting. This indicates that the integrated prior knowledge have had a negative impact on the network's guidance. From Figs. 10(b) and 11(b), it is evident that ResNet50's validation loss exhibited oscillations, possibly due to the residual structure introducing excessive redundant parameters, causing the network to learn unnecessary details. In contrast, MK-RNet, integrating melt pool prior knowledge, effectively mitigated ResNet50's oscillation issue and achieved a 0.65 % improvement in ACC and F1 score on the CMPQ dataset. This indicates that integrated prior knowledge had a positive effect on the network guidance. Additionally, compared to MobileNetV2, MK-MNet shows a decrease in ACC and F1 on the NIST dataset by 2.54 % and 2.42 %, respectively, but has a slight improvement on the CMPQ dataset, with increases of 0.87 % and 0.83 %. Overall, the guidance effect of prior knowledge varies significantly across different types of CNNs. For MK-DGNet and DGNet capturing deep data features through graph structures, MK-DGNet outperforms DGNet with an increase of 6.04 % and 6.17 % in ACC and F1 on the NIST dataset, rising from 79.52 % and 79.43 % to 85.56 % and 85.60 %, respectively. On the CMPQ dataset, MK-DGNet improves the ACC and F1 scores by 2.21 % and 2.17 %, respectively, increasing from 94.13 % and 94.20 % to 96.34 % and 96.37 %. Furthermore, the performance of DGNet on both datasets indicates that even without integrating prior knowledge, processing images using graph-based

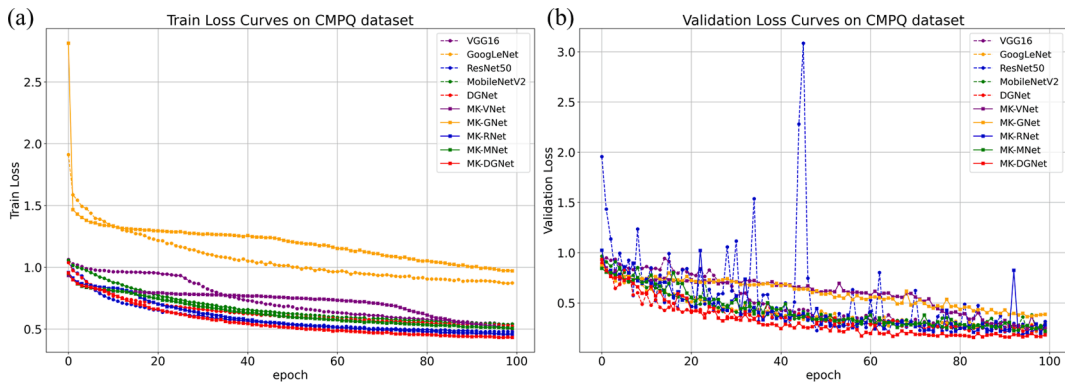**Fig. 10.** Loss curves of different models on NIST dataset. (a) Train Loss. (b) Validation Loss.

Fig. 11. Loss curves of different models on CMPQ dataset. (a) Train Loss. (b) Validation Loss.
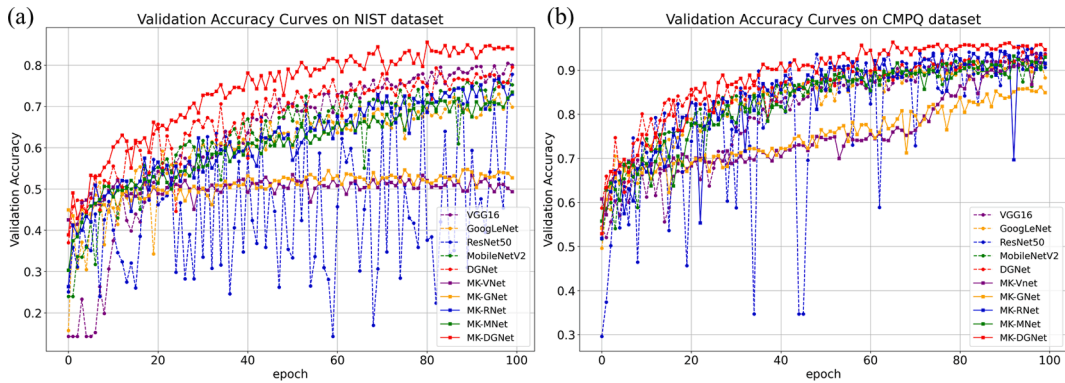

Fig. 12. Validation accuracy curves of different models. (a) NIST dataset. (b) CMPQ datasets.

methods can achieve results similar to traditional CNNs. These results not only demonstrate that viewing sensor data as graph structures is effective, but also highlight that the MK-DGNet framework is better at integrating prior knowledge than CNNs. Proper integration of prior knowledge is crucial for enhancing model performance.

Additionally, the working characteristic curve (ROC) of MK-DGNet and its area under the curve (AUC) are presented in Fig. 13. The ROC curve provides an intuitive understanding of the model's performance across different classes by showing the relationship between the true positive rate and false positive rate. Meanwhile, the AUC, as the area under the ROC curve, offers a concise metric for comprehensively evaluating the model's performance, with values closer to 1 indicating better performance. It can be observed that MK-DGNet achieves AUC values of 0.95 or higher for predicting seven classes on the NIST dataset. Specifically, for predicting the classes P100_V900 and P60_V25, the AUC reaches 1.00. The average AUC score is 0.97 on the NIST dataset. Moreover, on the CMPQ dataset for predicting the high, medium, and low quality of the melt pool, the AUC reaches 0.99. These results demonstrate that MK-DGNet exhibits robustness, accurately and reliably predicting the state and quality of melt pools during the laser additive manufacturing process.
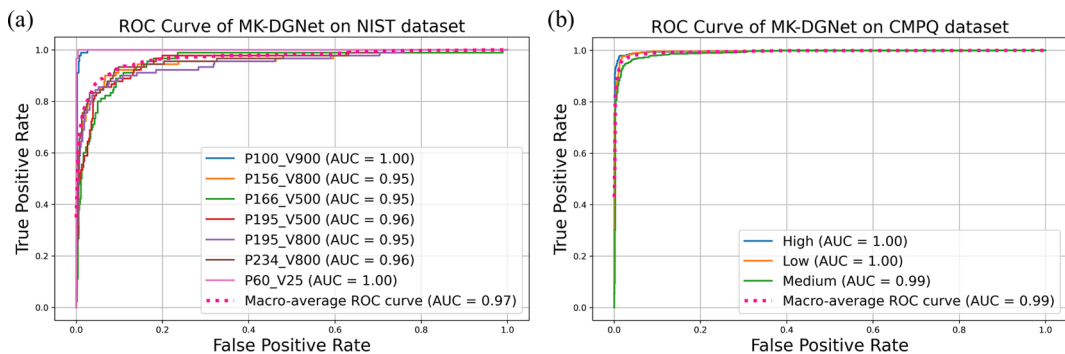

Fig. 13. ROC curve of MK-DGNet. (a) NIST dataset. (b) CMPQ datasets.

To further illustrate the effectiveness of incorporating prior knowledge, Figs. 14 and 15 display the confusion matrices of MK-DGNet and DGNet on NIST and CMPQ datasets. These matrices provide insights into the performance of the network classification models. Each row represents the actual class, while each column represents the predicted class by the model. For example, consider the first row in Fig. 15(a). The value 0.97 indicates that 97 % of samples in the P100_V900 category are correctly classified by the model. The last column in the same row shows 0.02, indicating that 2 % of samples in P100_V900 are incorrectly classified as P60_V25. An ideal classifier would produce a confusion matrix with values of 1 on the diagonal (representing 100 % correct classifications) and 0 elsewhere (indicating no misclassifications).

From Fig. 14, the confusion matrix indicates that MK-DGNet and DGNet achieve nearly 100 % accuracy on the P100_V900 and P60_V25 categories in the NIST dataset. For other categories, MK-DGNet shows significant improvements compared to DGNet. For other categories, MK-DGNet shows significant improvements compared to DGNet. Specifically, classification accuracies for P166 V500, P195 V500, and P195 V800 are improved by 10 %, 17 %, and 9 %, respectively. Additionally, P156 V800 and P234 V800 show modest improvements of 3 % and 4 %, respectively. For the CMPQ dataset, MK-DGNet also demonstrates a 1–3 % improvement in the classification accuracy for high, medium, and low melt pool quality categories, as shown in Fig. 15. The performance on both datasets suggests that the incorporation of prior knowledge effectively supplements the network's features, guiding and participating in network training.

4.2. Visualization

To further elucidate the effectiveness of incorporating prior knowledge and the importance of corresponding knowledge, we employ interpretable methods to decouple networks and elucidate the causal relationships between inputs and outputs of machine learning models. This is particularly essential for the development of online monitoring models for LPBF, as the relationship between process parameters and part quality is not yet fully understood [53]. Through interpretability methods, we can gain a deeper understanding of the model's perception capabilities, enhancing the reliability of deep learning approaches.

T-SNE is a popular high-dimensional data visualization method because it can map high-dimensional data into two- or three-dimensional space. We apply t-SNE to transform the NIST and CMPQ validation sets into two-dimensional space, and the results are depicted in Fig. 16. As shown in Fig. 16(a), the NIST raw data for P100_V900 and P60_V25 show a clustering trend, while other categories have significant overlap. This could be attributed to the fact that these categories represent transition processes, meaning they occur at the edges of parts, during laser steering, or at region boundaries, where the process conditions are relatively simple. Conversely, other categories involve various scanning strategies such as serpentine, back-and-forth, re-melt, and island patterns. These complex conditions lead to significant variations in melt pool, resulting in considerable overlap in the original data in the low-dimensional space. Fig. 16(b) indicates that although the CMPQ original data also show overlap in the low-dimensional space, different categories of samples have begun to exhibit clustering trend. This could be attributed to the fact that the variation in CMPQ melt pool quality is mainly caused by changes in P and V, with fewer influencing factors compared to the NIST dataset, resulting in lower complexity than NIST dataset.

The network utilized in this study can be summarized as comprising a feature extractor and a fully connected layer classifier. The feature extractor is employed to extract high-dimensional features of melt pool, while the fully connected layer maps these high-dimensional features to specified categories. To evaluate the effectiveness of different networks in feature extraction for quality monitoring, we utilized t-SNE to analyze the feature extractors of VGG16, GoogLeNet, ResNet50, and MK-DGNet. Specifically, we visualized the outputs of the feature extractors on the NIST and CMPQ test sets, which are the inputs to the fully connected layer. the results are depicted in Figs. 17 and 18. On the NIST dataset, the four models effectively separated the features of P100_V900 and P60_V25, confirming the confusion matrix results shown in Fig. 14(a) and the patterns illustrated in Fig. 16(a). However, as shown in Fig. 17(a, b, c), there remains significant overlap among other categories in the feature space of the three CNNs. In contrast, in the low-dimensional feature space of MK-DGNet as shown in Fig. 17(d), although some feature overlap among other categories still exists, the center distances of different categories are increasing, moving away from the overlapping region, and the samples show a tendency to
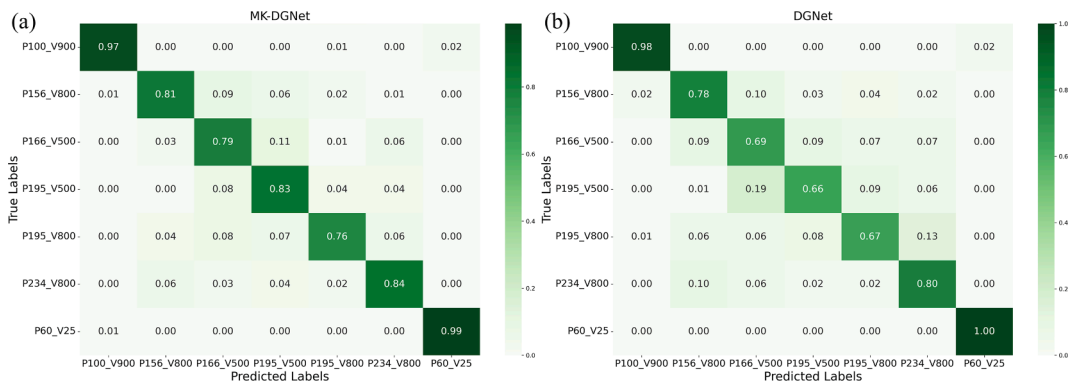

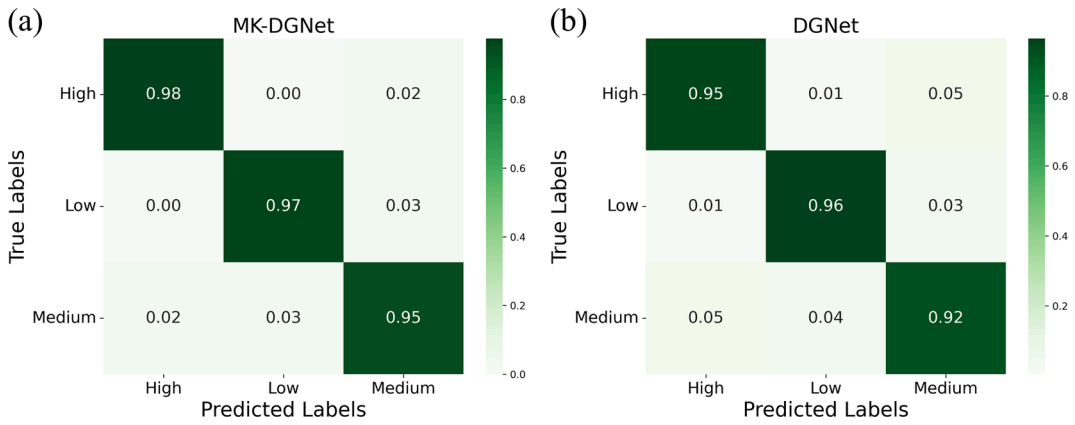**Fig. 14.** Confusion matrix on NIST dataset. (a)MK-DGNet. (b) DGNet.

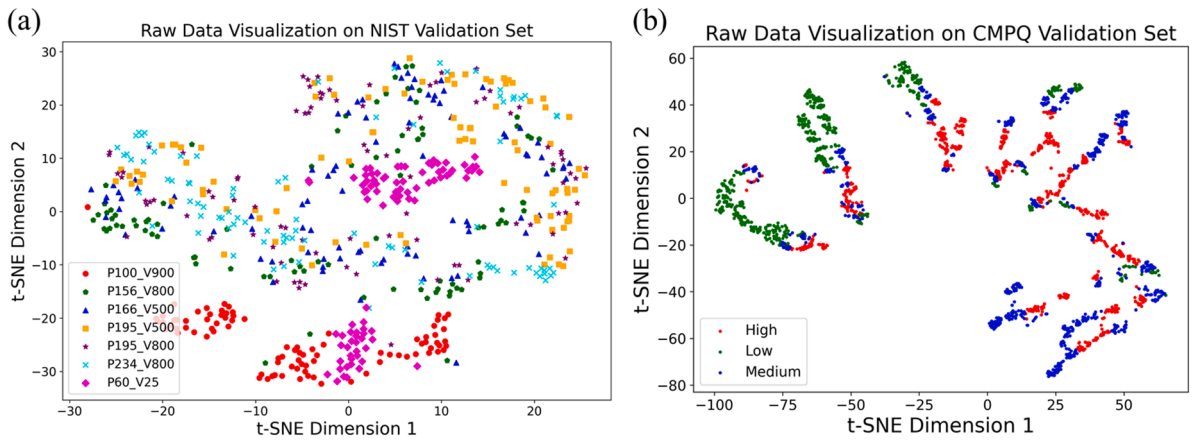**Fig. 15.** Confusion matrix on CMPQ dataset. (a)MK-DGNet. (b) DGNet.

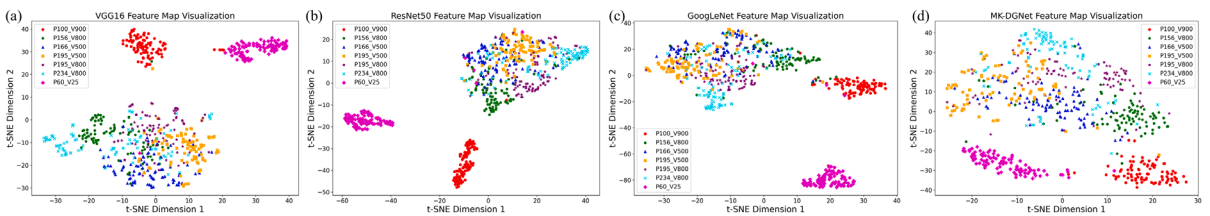**Fig. 16.** T-SNE Visualization on Raw Data. (a) NIST Validation Set. (b) CMPQ Validation Set.

Fig. 17. Different models T-SNE visualization on NIST Validation Set. (a) VGG16. (b) GoogLeNet. (c) ResNet50. (d) MK-DGNet.
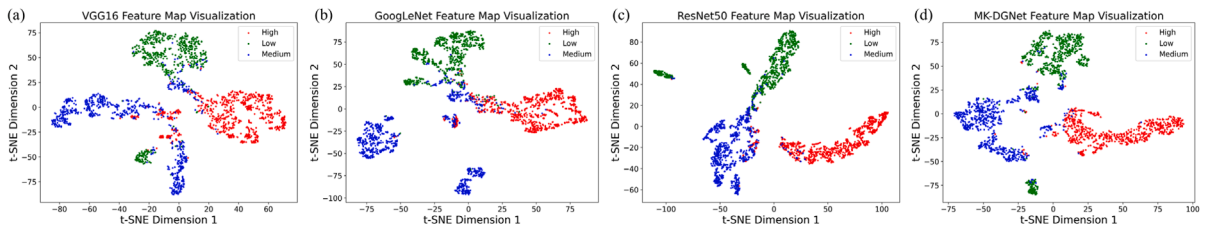

Fig. 18. Different models T-SNE visualization on CMPQ Validation Set. (a) VGG16. (b) GoogLeNet. (c) ResNet50. (d) MK-DGNet.

cluster around their respective centers. On the CMPQ dataset, although the different networks all show great feature space distribution, VGG16 and GoogLeNet still have a few overlapping features among different categories, while ResNet50 and MK-DGNet demonstrate clearer decision boundaries, as shown in Fig. 18. These results show that MK-DGNet outperforms these CNNs in feature extraction quality. The superior performance may be attributed to the fact that variations in process conditions impact the dynamic behavior of the melt pool, which affects the characteristics of melt pool in turn. MK-DGNet incorporate the characteristics of melt pool variations through prior knowledge of graph forms, improving feature extraction quality.

Furthermore, we employed another interpretability method to elucidate the relationship between network inputs and outputs. The Integrated Gradients method is a technique used to explain the predictions of deep learning models [54]. Integrated gradients measure the contribution of specific knowledge feature to the model's decision by calculating the accumulated gradients of the specific knowledge input to the model output. Integrated Gradients are defined as the path integral of gradients along a straight path from a baseline x' to the input x. The baseline is typically chosen as either the zero vector or an all-black image. The integrated gradient along the i-th dimension between the input x and the baseline x' is defined as follows, where $F(x)$ represents the output of the neural network model.

$$IntegratedGrads_i(x) ::= \left(x_i - x'_i\right) \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha \tag{19}$$

Integrated Gradients can provide insights into which input features have the most significant impact on the classification within a neural network. It offers direct causal explanations between inputs and outputs without requiring any prior knowledge of LPBF physics. This method facilitates an intuitive understanding of the model's decisions and helps explain the basis for predictions under specific inputs. By examining the score values for each output, we can intuitively assess the importance of the input knowledge in the network's decision-making process. Additionally, we can determine which components of the prior knowledge vector are most critical.

We employ the integrated gradient method to investigate the importance of pixel inputs in MK-DGNet. Through visualization of the integrated gradients of image pixels, we can understand which regions of the image the network focuses on. In Fig. 19, it is evident that regardless of the classification task on either the NIST or CMPQ dataset, MK-DGNet consistently emphasizes the melt pool region. This indicates that the network accomplishes the classification task by capturing the high-dimensional information of melt pool. Because melt pool encapsulates the high-dimensional information of the complex physical processes of the LPBF, this aligns with our conventional judgment. By analyzing melt pool region, we can better determine whether melt pool's status is ideal or abnormal, thereby providing more refined quality control. Additionally, this visualization can separate melt pool region from other regions, as shown in the NIST example image of Fig. 19, where the network's attention to non-melt pool region decreases. Overall, through the visualization of image integral gradients, we can highlight the importance of image pixels in network decisions, providing intuitive evidence for network decisions. This method not only helps us understand the decision-making process of the network in classification or other tasks but also reveals which pixel information in the input image is more critical to the network, thereby assisting us in further optimizing the network's performance and interpreting its behavior.

In Fig. 20, the average integrated gradients of melt pool prior knowledge vectors for different categories are illustrated. The absolute value of specific knowledge input's integral gradient reflects its contribution to a specific category predicted by the model. A larger absolute value indicates a stronger contribution. A positive value means that the specific knowledge has a positive contribution to the model's prediction for the specified category, increasing the probability of the model predicting that category. Conversely, a negative value indicates that the specific knowledge has a negative contribution, decreasing the probability of predicting that category. By examining the integral gradients of the prior knowledge vectors, several interesting observations can be made.

On the NIST and CMPQ datasets, the network exhibits differential attention to knowledge across different categories, with a particular emphasis on statistical knowledge. This indicates that statistical knowledge effectively reveals the distribution characteristics of the melt pool, uncovering information regarding stability, symmetry, and other factors relevant to identifying melt pool morphology and quality. Additionally, the network shows considerable attention to the physically intuitive characteristics of the melt pool, but the degree and correlation of attention to melt pool area, length, and width vary across different categories. On the CMPQ dataset, the network's attention to melt pool area surpasses its attention to melt pool length and width. This is because melt pool area better reflects changes in energy density. Insufficient energy density can lead to lack of fusion, while excessive energy density can lead to keyhole. Therefore, an appropriate range of energy density is crucial for ensuring part quality. Overall, the relationship between energy density and part quality resembles an inverted U-shaped curve, with melt pool area serving as a better indicator of this combined variation than melt pool length and width. As shown in Fig. 20(b), both low-quality and high-quality melt pools exhibit a negative correlation with melt pool area. This can be explained by the fact that in low-quality melt pools, a smaller melt pool area increases the probability of low-quality melt pools, thus leading to the observed negative correlation. In high-quality melt pools, larger melt pool areas are associated with higher energy density, which can result in decreased melt pool quality, hence the negative correlation observed. In the moderate melt pool quality range, as energy density gradually increases, melt pool area expands, ensuring thorough powder fusion and improving printing quality, thus resulting in a positive correlation between network attention and melt pool area. These observations align with our understanding of existing melt pool knowledge.

It's important to emphasize that MK-DGNet does not rely solely on any single knowledge for judgment but integrates various factors' influences into its decision-making process throughout network training. Here, integral gradients provide new insights into LPBF, demonstrating the basis of the network's decisions and indicating the positive or negative correlation between classification categories and knowledge. LPBF experts can use these correlations as a means to understand the complex physical processes of LPBF further and explore guiding principles.
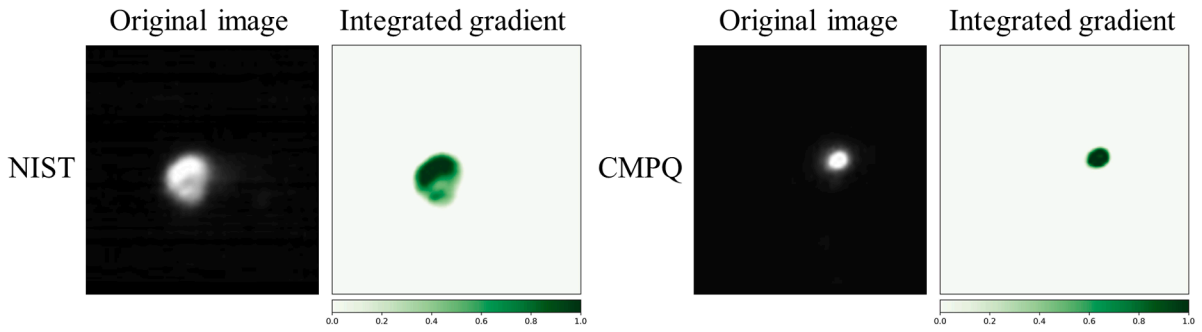
Fig. 19. Visualization of the integrated gradient of the melt pool image.
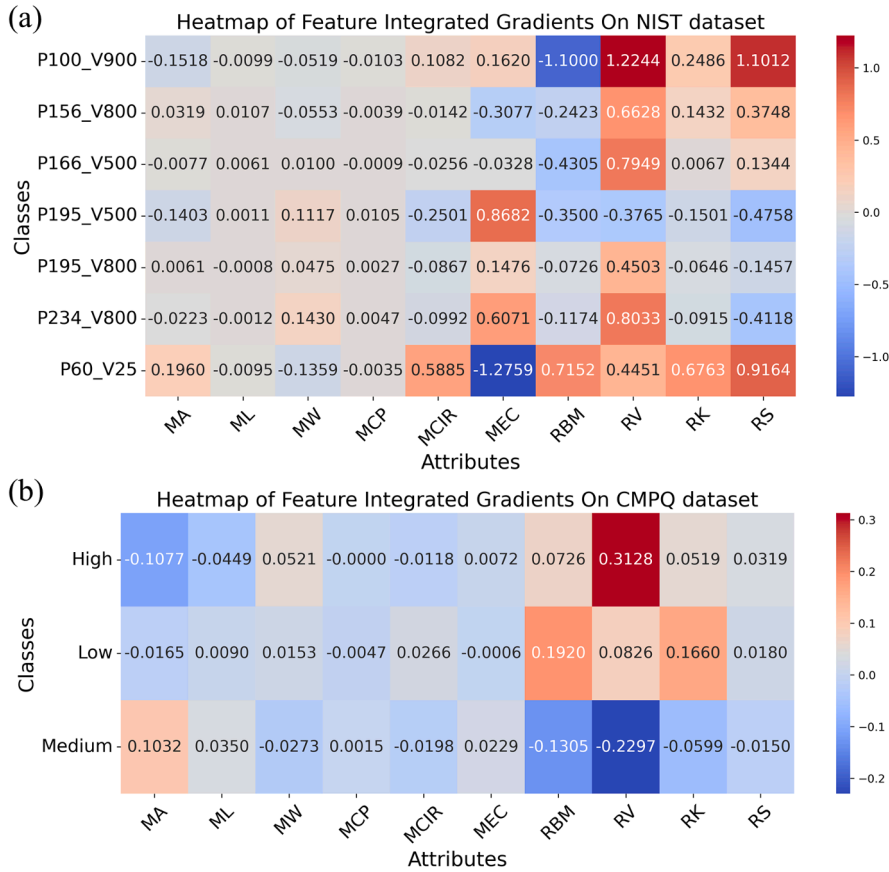

Fig. 20. Integrated gradient of melt pool prior knowledge vector. (a)NIST dataset. (b)CMPQ dataset.

4.3. Ablation experiment

As mentioned during the CMPQ dataset construction, merely using three labels might lead the model to find shortcut solutions, resulting in simplification. Therefore, in this section, we investigate the identification of melt pool variations under slight perturbations of process parameters to ensure that our results are not affected by model simplifications. Similar to the label in the NIST dataset, we created a 98-class classification task based on different combinations of process parameters as described in Section 2.2.1, according to the forming results of the parts. Specifically, for each category, 200 melt pool images were used as training samples, which were in constructing the CMPQ dataset. Additionally, 50 new images were added for each category as test samples, which were not present in the CMPQ dataset. Ultimately, we created the verification dataset, termed the coaxial melt pool quality verification (CMPQ-V) dataset, with 98 classes classification task where the training set contains 19,600 samples, and the validation set contains 4,900 samples.

Fig. 21 displays the classification performance of CNNs and MK-DGNet on CMPQ-V dataset. MK-DGNet still achieves the highest

ACC and F1 on the CMPQ-V dataset, with values of 85.56 % and 85.60 %. The trend of validation accuracy variation remains consistent with the CMPQ dataset. In CNNs, except for the large-parameter VGG16, other CNN models exhibit varying degrees of degradation. These results demonstrate that MK-DGNet has a favorable transferability and is not affected by model simplification.

5. Conclusion

In this study, we propose a novel machine learning architecture called MK-DGNet, aimed at integrating prior physical knowledge and employing a more versatile graph data format for application in in-situ quality monitoring of LPBF. The effectiveness and robustness of the developed framework were validated using publicly available NIST datasets and internally created CMPQ datasets, employing explainable artificial intelligence methods to validate the efficacy of prior knowledge. The main findings are as follows:

(1) The developed MK-DGNet achieved the highest ACC and F1 scores on both datasets. MK-DGNet demonstrated rapid accuracy improvement and loss reduction during the initial stages of training. After 20 epochs, its accuracy became significantly outstanding and maintained this highest level in most subsequent epochs compared to other models. The average AUC scores of MK-DGNet on the NIST and CMPQ datasets were 0.97 and 0.99, respectively, demonstrating its robustness. Ablation experiments indicate that our results are not affected by model simplification and show good transferability. The proposed deep graph neural network guided by prior knowledge can effectively handle visual tasks and has the potential to become an essential component of LPBF systems.

(2) The proposed approach, incorporating prior knowledge with raw sensor data and representing the data in the form of graph, is feasible and effective. The ability of deep graph networks to integrate prior knowledge surpasses CNNs. On the NIST dataset, the designed MK-DGNet achieved classification accuracy improvements of 10 %, 17 %, and 9 % for P166 V500, P195 V500, and P195 V800, respectively. Additionally, P156_V800 and P234_V800 showed moderate improvements of 3 % and 4 %, respectively. On the CMPQ dataset, classification accuracy for high, medium, and low melt pool quality categories improved by 1–3 %.

(3) Explainable artificial intelligence methods elucidate the basis of network decisions and the effectiveness and importance of the incorporated prior knowledge. The t-SNE visualization indicate that MK-DGNet extracts higher quality features compared to CNNs. Attribution score visualization show that MK-DGNet focuses on the melt pool area in images for decision-making and that the network's attention to different categories of knowledge varies. By explaining the relationship between melt pool morphology and melt pool quality enhance the network's interpretability and reliability.

In our future work, we plan to expand our knowledge scope to include principles of physics and engineering. We aim to construct a defect knowledge graph for LPBF and integrate it into our developing prior-knowledge-guided deep graph network framework. This integration will enable more reliable online monitoring of LPBF processes.

CRediT authorship contribution statement

Yingjie Zhang: Supervision, Resources, Funding acquisition, Conceptualization. **Honghong Du:** Writing – review & editing, Writing – original draft, Methodology. **Kai Zhao:** Supervision, Project administration. **Jiali Gao:** Supervision, Project administration. **Xiaojun Peng:** Investigation, Data curation. **Lang Cheng:** Validation, Resources, Project administration. **Canneng Fang:** Software, Formal analysis. **Gang Chen:** Validation, Resources, Project administration.
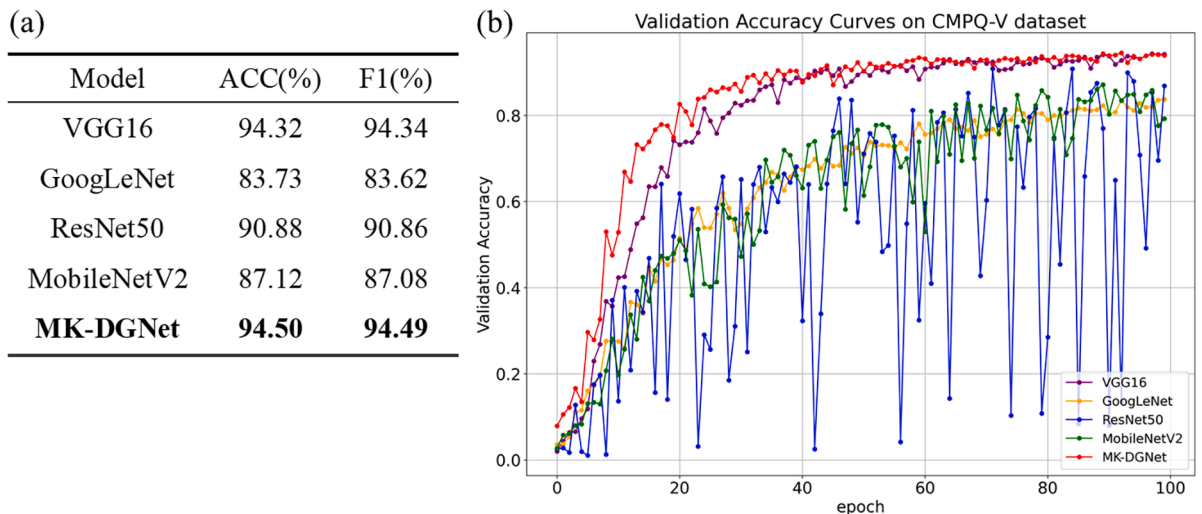
(a)

Model	ACC(%)	F1(%)
VGG16	94.32	94.34
GoogLeNet	83.73	83.62
ResNet50	90.88	90.86
MobileNetV2	87.12	87.08
MK-DGNet	**94.50**	**94.49**

Fig. 21. Classification performance on CMPQ-V dataset. (a) ACC and F1. (b) Validation Accuracy.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgement

References

[1] I. Gibson, D. Rosen, B. Stucker, M. Khorasani, in: Additive Manufacturing Technologies, Springer International Publishing, Cham, 2021, https://doi.org/10.1007/978-3-030-56127-7.

[2] W.E. King, A.T. Anderson, R.M. Ferencz, N.E. Hodge, C. Kamath, S.A. Khairallah, A.M. Rubenchik, Laser powder bed fusion additive manufacturing of metals; physics, computational, and materials challenges, Appl. Phys. Rev. 2 (2015) 041304, https://doi.org/10.1063/1.4937809.

[3] N. Haghdadi, M. Laleh, M. Moyle, S. Primig, Additive manufacturing of steels: a review of achievements and challenges, J. Mater. Sci. 56 (2021) 64–107, https://doi.org/10.1007/s10853-020-05109-0.

[4] T. DebRoy, T. Mukherjee, J.O. Milewski, J.W. Elmer, B. Ribic, J.J. Blecher, W. Zhang, Scientific, technological and economic issues in metal printing and their solutions, Nat. Mater. 18 (2019) 1026–1032, https://doi.org/10.1038/s41563-019-0408-2.

[5] C.Y. Yap, C.K. Chua, Z.L. Dong, Z.H. Liu, D.Q. Zhang, L.E. Loh, S.L. Sing, Review of selective laser melting: materials and applications, Appl. Phys. Rev. 2 (2015) 041101, https://doi.org/10.1063/1.4935926.

[6] D. Gu, Y.-C. Hagedorn, W. Meiners, G. Meng, R.J.S. Batista, K. Wissenbach, R. Poprawe, Densification behavior, microstructure evolution, and wear performance of selective laser melting processed commercially pure titanium, Acta Mater. 60 (2012) 3849–3860, https://doi.org/10.1016/j.actamat.2012.04.006.

[7] A. Du Plessis, Effects of process parameters on porosity in laser powder bed fusion revealed by X-ray tomography, Addit. Manuf. 30 (2019) 100871, https://doi.org/10.1016/j.addma.2019.100871.

[8] G.K.L. Ng, A.E.W. Jarfors, G. Bi, H.Y. Zheng, Porosity formation and gas bubble retention in laser metal deposition, Appl. Phys. A 97 (2009) 641–649, https://doi.org/10.1007/s00339-009-5266-3.

[9] Z. Ren, L. Gao, S.J. Clark, K. Fezzaa, P. Shevchenko, A. Choi, W. Everhart, A.D. Rollett, L. Chen, T. Sun, Machine learning–aided real-time detection of keyhole pore generation in laser powder bed fusion, Science 379 (2023) 89–94, https://doi.org/10.1126/science.add4667.

[10] C. Zhao, N.D. Parab, X. Li, K. Fezzaa, W. Tan, A.D. Rollett, T. Sun, Critical instability at moving keyhole tip generates porosity in laser melting, Science 370 (2020) 1080–1086, https://doi.org/10.1126/science.abd1587.

[11] S.M.H. Hojjatzadeh, N.D. Parab, Q. Guo, M. Qu, M. Xiong, C. Zhao, L.I. Escano, K. Fezzaa, W. Everhart, T. Sun, L. Chen, Direct observation of pore formation mechanisms during LPBF additive manufacturing process and high energy density laser welding, Int. J. Mach. Tool Manuf. 153 (2020) 103555, https://doi.org/10.1016/j.ijmachtools.2020.103555.

[12] S. Cooke, K. Ahmadi, S. Willerth, R. Herring, Metal additive manufacturing: technology, metallurgy and modelling, J. Manuf. Process. 57 (2020) 978–1003, https://doi.org/10.1016/j.jmapro.2020.07.025.

[13] Y. Zhang, G.S. Hong, D. Ye, K. Zhu, J.Y.H. Fuh, Extraction and evaluation of melt pool, plume and spatter information for powder-bed fusion AM process monitoring, Mater. Des. 156 (2018) 458–469, https://doi.org/10.1016/j.matdes.2018.07.002.

[14] H. Xu, H. Huang, In situ monitoring in laser melt injection based on fusion of infrared thermal and high-speed camera images, J. Manuf. Process. 92 (2023) 466–478, https://doi.org/10.1016/j.jmapro.2023.02.059.

[15] Q. Wang, X. Lin, X. Duan, R. Yan, J.Y.H. Fuh, K. Zhu, Gaussian process classification of melt pool motion for laser powder bed fusion process monitoring, Mech. Syst. Sig. Process. 198 (2023) 110440, https://doi.org/10.1016/j.ymssp.2023.110440.

[16] J. Yuan, H. Liu, W. Liu, F. Wang, S. Peng, A method for melt pool state monitoring in laser-based direct energy deposition based on DenseNet, Measurement 195 (2022) 111146, https://doi.org/10.1016/j.measurement.2022.111146.

[17] O. Kwon, H.G. Kim, M.J. Ham, W. Kim, G.-H. Kim, J.-H. Cho, N.I. Kim, K. Kim, A deep neural network for classification of melt-pool images in metal additive manufacturing, J. Intell. Manuf. 31 (2020) 375–386, https://doi.org/10.1007/s10845-018-1451-6.

[18] J. Li, L. Cao, H. Liu, Q. Zhou, X. Zhang, M. Li, Imbalanced data generation and fusion for in-situ monitoring of laser powder bed fusion, Mech. Syst. Sig. Process. 199 (2023) 110508, https://doi.org/10.1016/j.ymssp.2023.110508.

[19] F.-L. Fan, J. Xiong, M. Li, G. Wang, On interpretability of artificial neural networks: a survey, IEEE Trans. Radiat. Plasma Med. Sci. 5 (2021) 741–760, https://doi.org/10.1109/TRPMS.2021.3066428.

[20] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nat. Rev. Phys. 3 (2021) 422–440, https://doi.org/10.1038/s42254-021-00314-5.

[21] J. Willard, X. Jia, S. Xu, M. Steinbach, V. Kumar, Integrating scientific knowledge with machine learning for engineering and environmental systems, ACM Comput. Surv. 55 (2023) 1–37, https://doi.org/10.1145/3514228.

[22] M. Russell, P. Wang, Physics-informed deep learning for signal compression and reconstruction of big data in industrial condition monitoring, Mech. Syst. Sig. Process. 168 (2022) 108709, https://doi.org/10.1016/j.ymssp.2021.108709.

[23] Z. Wang, Z. Zhou, W. Xu, C. Sun, R. Yan, Physics informed neural networks for fault severity identification of axial piston pumps, J. Manuf. Syst. 71 (2023) 421–437, https://doi.org/10.1016/j.jmsy.2023.10.002.

[24] Q. Ni, J.C. Ji, B. Halkon, K. Feng, A.K. Nandi, Physics-Informed Residual Network (PIResNet) for rolling element bearing fault diagnostics, Mech. Syst. Sig. Process. 200 (2023) 110544, https://doi.org/10.1016/j.ymssp.2023.110544.

[25] M. Yin, S. Zhuo, L. Xie, L. Chen, M. Wang, G. Liu, Online monitoring of local defects in robotic laser additive manufacturing process based on a dynamic mapping strategy and multibranch fusion convolutional neural network, J. Manuf. Syst. 71 (2023) 494–503, https://doi.org/10.1016/j.jmsy.2023.10.005.

[26] S. Guo, C. Zamiela, L. Bian, Knowledge-transfer-enabled porosity prediction for new part geometry in laser metal deposition, J. Manuf. Process. 103 (2023) 64–77, https://doi.org/10.1016/j.jmapro.2023.08.002.

[27] Y. Lu, Y. Wang, L. Pan, A feature-based physics-constrained active dictionary learning scheme for image-based additive manufacturing process monitoring, J. Manuf. Process. 103 (2023) 261–273, https://doi.org/10.1016/j.jmapro.2023.08.040.

[28] R. Liu, S. Liu, X. Zhang, A physics-informed machine learning model for porosity analysis in laser powder bed fusion additive manufacturing, Int. J. Adv. Manuf. Technol. 113 (2021) 1943–1958, https://doi.org/10.1007/s00170-021-06640-3.

[29] Q. Zhu, Z. Liu, J. Yan, Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks, Comput. Mech. 67 (2021) 619–635, https://doi.org/10.1007/s00466-020-01952-9.

[30] Y. Du, T. Mukherjee, T. DebRoy, Physics-informed machine learning and mechanistic modeling of additive manufacturing to reduce defects, Appl. Mater. Today 24 (2021) 101123, https://doi.org/10.1016/j.apmt.2021.101123.

[31] H. Ko, Y. Lu, Z. Yang, N.Y. Ndiaye, P. Witherell, A framework driven by physics-guided machine learning for process-structure-property causal analytics in additive manufacturing, J. Manuf. Syst. 67 (2023) 213–228, https://doi.org/10.1016/j.jmsy.2022.09.010.

[32] K. Han, Y. Wang, J. Guo, Y. Tang, E. Wu, Vision GNN: An Image is Worth Graph of Nodes, 2022. http://arxiv.org/abs/2206.00272 (accessed April 24, 2024).

[33] T. Li, Z. Zhou, S. Li, C. Sun, R. Yan, X. Chen, The emerging graph neural networks for intelligent fault diagnostics and prognostics: a guideline and a benchmark study, Mech. Syst. Sig. Process. 168 (2022) 108653, https://doi.org/10.1016/j.ymssp.2021.108653.

[34] D. Zhang, E. Stewart, M. Entezami, C. Roberts, D. Yu, Intelligent acoustic-based fault diagnosis of roller bearings using a deep graph convolutional network, Measurement 156 (2020) 107585, https://doi.org/10.1016/j.measurement.2020.107585.

[35] C. Li, L. Mo, R. Yan, Rolling bearing fault diagnosis based on horizontal visibility graph and graph neural networks, in: 2020 International Conference on Sensing, Measurement & Data Analytics in the Era of Artificial Intelligence (ICSMD), IEEE, Xi'an, China, 2020, pp. 275–279, https://doi.org/10.1109/ICSMD50554.2020.9261687.

[36] M. Mozaffar, S. Liao, H. Lin, K. Ehmann, J. Cao, Geometry-agnostic data-driven thermal modeling of additive manufacturing processes using graph neural networks, Addit. Manuf. 48 (2021) 102449, https://doi.org/10.1016/j.addma.2021.102449.

[37] S. Zhu, Y. Zhang, B. Zhu, J. Zhang, Y. He, W. Xu, High cycle fatigue life prediction of titanium alloys based on a novel deep learning approach, Int. J. Fatigue 182 (2024) 108206, https://doi.org/10.1016/j.ijfatigue.2024.108206.

[38] B. Jiang, Y. Wang, H. Niu, X. Cheng, P. Zhao, J. Bao, GNNs for mechanical properties prediction of strut-based lattice structures, Int. J. Mech. Sci. 269 (2024) 109082, https://doi.org/10.1016/j.ijmecsci.2024.109082.

[39] B. Lane, H. Yeung, Process monitoring dataset from the additive manufacturing metrology testbed (AMMT): "three-dimensional scan strategies", J. Res. Natl. Inst. Stan. 124 (2019) 124033 https://doi.org/10.6028/jres.124.033.

[40] S. Clijsters, T. Craeghs, S. Buls, K. Kempen, J.-P. Kruth, In situ quality control of the selective laser melting process using a high-speed, real-time melt pool monitoring system, Int. J. Adv. Manuf. Technol. 75 (2014) 1089–1101, https://doi.org/10.1007/s00170-014-6214-8.

[41] Y. Cai, J. Xiong, H. Chen, G. Zhang, A review of in-situ monitoring and process control system in metal-based laser additive manufacturing, J. Manuf. Syst. 70 (2023) 309–326, https://doi.org/10.1016/j.jmsy.2023.07.018.

[42] L. Chen, G. Bi, X. Yao, J. Su, C. Tan, W. Feng, M. Benakis, Y. Chew, S.K. Moon, In-situ process monitoring and adaptive quality enhancement in laser additive manufacturing: a critical review, J. Manuf. Syst. 74 (2024) 527–574, https://doi.org/10.1016/j.jmsy.2024.04.013.

[43] J. Li, L. Cao, J. Xu, S. Wang, Q. Zhou, In situ porosity intelligent classification of selective laser melting based on coaxial monitoring and image processing, Measurement 187 (2022) 110232, https://doi.org/10.1016/j.measurement.2021.110232.

[44] A. Gaikwad, R.J. Williams, H. De Winton, B.D. Bevans, Z. Smoqi, P. Rao, P.A. Hooper, Multi phenomena melt pool sensor data fusion for enhanced process monitoring of laser powder bed fusion additive manufacturing, Mater. Des. 221 (2022) 110919, https://doi.org/10.1016/j.matdes.2022.110919.

[45] Z. Smoqi, A. Gaikwad, B. Bevans, M.H. Kobir, J. Craig, A. Abul-Haj, A. Peralta, P. Rao, Monitoring and prediction of porosity in laser powder bed fusion using physics-informed meltpool signatures and machine learning, J. Mater. Process. Technol. 304 (2022) 117550, https://doi.org/10.1016/j.jmatprotec.2022.117550.

[46] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2021. http://arxiv.org/abs/2010.11929 (accessed July 22, 2024).

[47] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, P.S. Yu, A comprehensive survey on graph neural networks, IEEE Trans. Netw. Learning Syst. 32 (2021) 4–24, https://doi.org/10.1109/TNNLS.2020.2978386.

[48] G. Li, M. Muller, A. Thabet, B. Ghanem, DeepGCNs: Can GCNs Go As Deep As CNNs?, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, Seoul, Korea (South), 2019: pp. 9266–9275. https://doi.org/10.1109/ICCV.2019.00936.

[49] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, (2015). http://arxiv.org/abs/1409.1556 (accessed April 28, 2024).

[50] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going Deeper with Convolutions, (2014). http://arxiv.org/abs/1409.4842 (accessed April 28, 2024).

[51] K. He, X. Zhang, S. Ren, J. Sun, in: Deep Residual Learning for Image Recognition, IEEE, Las Vegas, NV, USA, 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.

[52] M. Tan, Q.V. Le, EfficientNetV2: Smaller Models and Faster Training, (2021). http://arxiv.org/abs/2104.00298 (accessed April 28, 2024).

[53] M.M. Sato, V.W.H. Wong, K.H. Law, H. Yeung, P. Witherell, Explainability of laser powder bed fusion melt pool classification using deep learning, (2023).

[54] M. Sundararajan, A. Taly, Q. Yan, Axiomatic Attribution for Deep Networks, (2017). http://arxiv.org/abs/1703.01365 (accessed April 24, 2024).