



# Material-Adaptive Anomaly Detection Using Property-Concatenated Transfer Learning in Wire Arc Additive Manufacturing

Seung-Jun Shin<sup>1</sup> · Ju-Hong Lee<sup>2</sup> · Sainand Jadhav<sup>3</sup> · Duck Bong Kim<sup>4</sup>

Received: 5 April 2023 / Revised: 16 October 2023 / Accepted: 18 October 2023  
© The Author(s), under exclusive licence to Korean Society for Precision Engineering 2023

## Abstract

Wire arc additive manufacturing is a promising additive manufacturing process because of its high deposition rate, and material diversity. However, the low quality of melted parts is a critical issue, owing to the difficulty in establishing design rules for process–structure–property–performance. Previous studies have resolved this challenge by deriving anomaly detection models for quality monitoring and have largely relied on machine learning by training melt pool image data. Acquiring sufficient data is a key to obtaining reliable models in machine learning; however, an issue arises from concerning the cost intensiveness in high-cost materials. We propose a material-adaptive anomaly detection method to detect balling defects in a target material using property-concatenated transfer learning. First, transfer learning is applied to derive convolutional neural network (CNN)-based models from a source material and transfer them to a target material, wherein data are insufficient and machine learning rarely achieves high performance. Second, material properties are concatenated on transfer learning as additional features onto image features, contrary to typical transfer learning where CNNs only extract image features. We perform experiments in a gas tungsten arc welding system with low-carbon steel (LCS), stainless steel (STS), and inconel (INC) materials. Our models achieve best classification accuracies of 82.95%, 89.47%, and 84.22% when transferring from LCS to STS, LCS to INC, and STS to INC, respectively, compared with 78.03%, 86.37%, and 73.63% obtained using typical transfer learning. The proposed method can effectively resolve the data scarcity by model transfer from sufficient datasets in low-cost materials to rare datasets in high-cost materials. Moreover, it outperforms typical transfer learning because material properties are learned as manufacturing-knowledge features, accounting for melting and hardening characteristics of materials.

**Keywords** Wire arc additive manufacturing · Anomaly detection · Transfer learning · Convolutional neural network · Quality monitoring · Material property

## 1 Introduction

Additive manufacturing (AM) is a process of joining materials to produce parts layer upon layer based on three-dimensional (3D) model data, contrary to subtractive and formative manufacturing [1]. AM has been categorized into seven processes: binder jetting, material jetting, powder bed fusion, sheet lamination, vat photopolymerization (VP), material extrusion (ME), and direct energy deposition (DED) [1]. Further, non-metal AM involves binder jetting, material jetting, VP, and ME; whereas metal AM includes powder bed fusion, sheet lamination, and DED [2]. Non-metal AM mostly involves using polymers as feedstock materials. VP can fabricate lightweighted and high energy absorbed parts using penetrated and interpenetrated cellular lattice structures [3]. ME is known as the highest-coefficient

---

✉ Duck Bong Kim  
dkim@nttech.edu

<sup>1</sup> School of Interdisciplinary Industrial Studies, Hanyang University, Seoul, Republic of Korea

<sup>2</sup> Department of Industrial Data Engineering, Hanyang University, Seoul, Republic of Korea

<sup>3</sup> Department of Mechanical Engineering, Tennessee Technological University, Cookeville, TN 38505, USA

<sup>4</sup> Department of Manufacturing and Engineering Technology, Tennessee Technological University, Cookeville, TN 38505, USA

and widely-diffused process among non-metal AM owing to its low equipment cost and fast build-up rates, as it extrudes material in a semisolid state via a nozzle and solidifies the extruded material [4]. Meanwhile, metal AM has exhibited flexibility in the component geometries and designs of metallic parts because it can fuse and solidify metal alloy structures on a substrate owing to the supplement of high-density energy. In the metal AM, DED can be divided into powder bed, powder feed, and wire feed processes in terms of feedstock materials or into laser beam, electron beam and arc-based systems in terms of energy sources. The latter is known as wire arc additive manufacturing (WAAM), which is a process of melting a wire feedstock and depositing the part layer upon layer by using an arc-based system as a source of energy.

WAAM consists of a wire as the feeding stock, a welding arc as the energy source, and a robot arm as the deposition operator. WAAM can be categorized into gas metal arc welding, plasma arc welding and gas tungsten arc welding (GTAW) based on the heat sources [5]. WAAM possesses the benefits of a high deposition rate, near-net-shape fabrication, diversity in applicable wires, cost efficiency for large parts owing to its low-cost equipment installation, and less material waste owing to the low buy-to-fly ratio [6]. Considering these benefits, WAAM has been applied in the automotive, aerospace, and shipbuilding industries [7]. These benefits are similar to those of ME as mentioned above. However, their differences occur in terms of feedstock materials because WAAM treats metallic parts although ME mostly deals with non-metallic parts, e.g., acrylonitrile butadiene styrene and polylactic acid.

Known defects in WAAM include balling, porosity, deformation, oxidation, delamination, cracking, high residual stress, and low surface finish [8]. These defects result in low precision, poor surface quality, and deterioration of mechanical properties [9]. The layer-by-layer stacking mechanism causes poor dimensional accuracy and surface finish and thus leads to make a volumetric error between the designed and fabricated parts, which has limited widespread applications of additive manufacturing [10]. Moreover, WAAM can cause defects and undesirable features, e.g., heterogeneous microstructures, owing to non-equilibrium thermal cycles and induce process instability, i.e., balling formation and spatter, adversely affecting surface roughness and mechanical properties. Hence, methods to achieve process stability and high-quality parts are required in WAAM. These methods require a solid understanding of the underlying physics together with formulation of mathematical and statistical models. The WAAM community is actively seeking data-driven solutions for monitoring and detecting defects based on in-situ and real-time approaches [11]. Anomaly detection can be used as a primary solution for detecting abnormalities during the process. Anomaly detection uses sensor data

to identify patterns that did not conform to a well-defined notion of normal behavior [12]. Anomalies incur defects and thus have to be automatically detected to assure the quality of products and reduce the cost of post-process treatment [13].

The sensor data are numerically analyzed and transformed into mathematical models for anomaly detection. For this, the design of experiments (DOE) can be used to effectively determine the numerical relationship between the input and output data with a small set of experiments [14]. However, the DOE is valid under restricted experimental conditions and is not appropriate for real-time monitoring and control. Recently, machine learning has gained increasing attention as a data-driven approach for overcoming these limitations. Machine learning derives mathematical models for making decisions based on training data acquired from experiments. Machine learning can generate machine-specific models for real-time monitoring and control in dynamic environments [15]. Machine learning can be classified into supervised, unsupervised, and reinforcement learning [16]. In supervised learning, human observations are used to identify input and output variables based on expertise and knowledge. Computers learn numerical causality between these two variables using a training dataset, wherein every input datum is labeled with a corresponding output value [17]. On the one hand, unsupervised learning makes inferences from unlabeled data by exploring hidden patterns or grouping similar data clusters from a dataset [18]. On the other hand, reinforcement learning is a semi-supervised model that interacts with its environment and learns to act optimally to gain the most significant reward [19].

Considering the advantages of machine learning, the AM community has been characterizing the linkage among process–structure–property–performance as the *design rule* [9]. However, it is challenging to establish this relationship because machine learning requires massive data to achieve reliable results. For example, the materials and process parameters significantly affect the corresponding microstructures and quality performance. Consequently, their combinations incur an exponentially increasing number of data samples. This effect is termed as the *curse of dimensionality* [20]. While considering a high-performance material (e.g., Inconel 625 or Ti-6Al-4V), which is expensive, obtaining sufficient datasets becomes challenging. Moreover, data missing and scarcity occasionally occur owing to the complexity of the process and dynamics of manufacturing environments [21]. Defects should be minimized and detected in-process to minimize post-process treatment and reduce product disposal. Therefore, a cost-effective method is required for real-time monitoring and control.

Transfer learning (TL) represents a key solution to this problem. TL is a learning approach that aims to extract knowledge from source domains or tasks to be used for

a target domain or task [22]. In AM, TL can be used to extract original features from raw image or numerical data in a source material and then transfer and adjust the features to create anomaly detection models applicable to the target material. TL exhibits good accuracy especially when the features learned from a source material possess high *transferability*, which signifies whether features are correctly learned to obtain generic phenomena across domains. TL can be cost-effective when source materials and source data are inexpensive, the target material is expensive, and the target data are insufficient or expensive. Although this TL approach can facilitate the establishment of design rules, it has not been comprehensively investigated, and related knowledge is lacking.

This study proposes a TL-based material-adaptive anomaly detection method to use data inexpensively obtained using a GTAW-based process. The proposed method generates anomaly detection models for classifying balling defects as *abnormal* based on property-concatenated TL. The proposed method uses TL to derive CNN-based anomaly detection models by transferring models derived from a source material to a target material, where machine learning rarely achieves good accuracy owing to data scarcity. In addition, the proposed method applies the property concatenation to combine material properties as additional features onto image features, contrary to typical TL where CNNs use original image features extracted from source material data. This property concatenation aims to reflect the knowledge of WAAM, i.e., thermal properties of materials affect melting and solidification mechanisms during deposition, thereby improving accuracy. The proposed method uses CNNs to apply image-based learning for time-series classification by extracting the image features from voltage image snapshots, which are converted and segregated from a time-series profile of numerical voltage data. Experiments are performed in a gas tungsten arc welding system to demonstrate the feasibility and validity of the proposed method

with three materials: low-carbon steel (LCS), stainless steel 316L (STS), and Inconel 625 (INC).

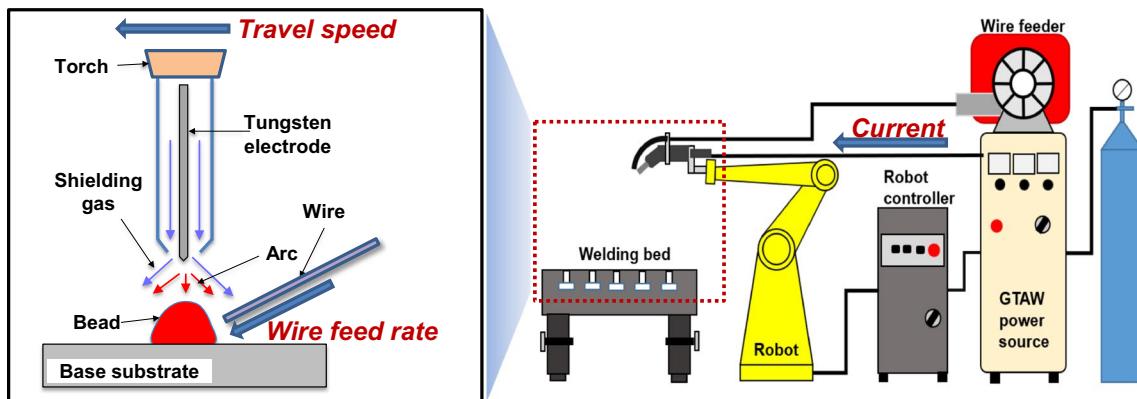
This study is organized as follows. Section 2 introduces related works; Sect. 3 explains the experiments; Sect. 4 proposes the method; Sect. 5 describes and discusses the validation of the proposed method; Sect. 6 concludes the study.

## 2 Related Works

### 2.1 Wire Arc Additive Manufacturing and Balling Defect

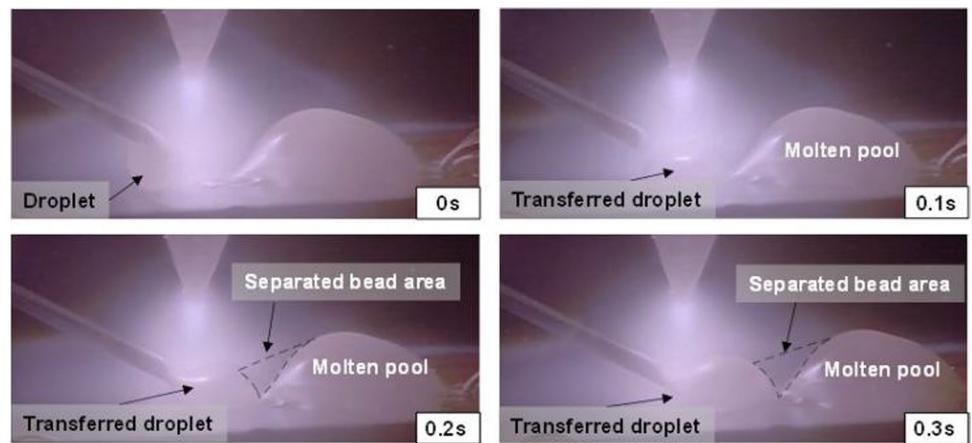
WAAM uses DED as an energy source to fabricate 3D metallic parts. The feedstock materials currently available in the welding industry are titanium, aluminum, steel, nickel, and inconel alloys. Figure 1 illustrates a schematic diagram of GTAW and its WAAM system. GTAW helps create weld beads on a base substrate using an arc generated from a non-consumable tungsten electrode. The arc generates a molten pool, solidifying the metallurgical bond between the feedstock and base substrate (or previously deposited layer). In GTAW, the primary process parameters are current, travel speed (TS), and wire feed rate (WFR). The current affects the intensity of the heat input, while the TS and WFR influence the dynamics of the bead formation. They are determined by operators and are mutually independent because their settings are separately applied to the control of associated devices, including a power source, torch, and wire feeder.

Figure 2 shows the balling phenomenon. Balling can be specified as an irregular bead surface contour comprising protrusions caused by the separation of spherical droplets [23]. The filler metal starts melting at the wire tip and forms a droplet at the center of the arc. Droplets are formed sequentially, and molten pools are shaped on the base substrate as the filler metal continues to move under the arc. The



**Fig. 1** Schematic diagram of gas tungsten arc welding (left) and wire arc additive manufacturing system (right)

**Fig. 2** Formation of the balling phenomenon (irregular bead shape and spherical droplets)



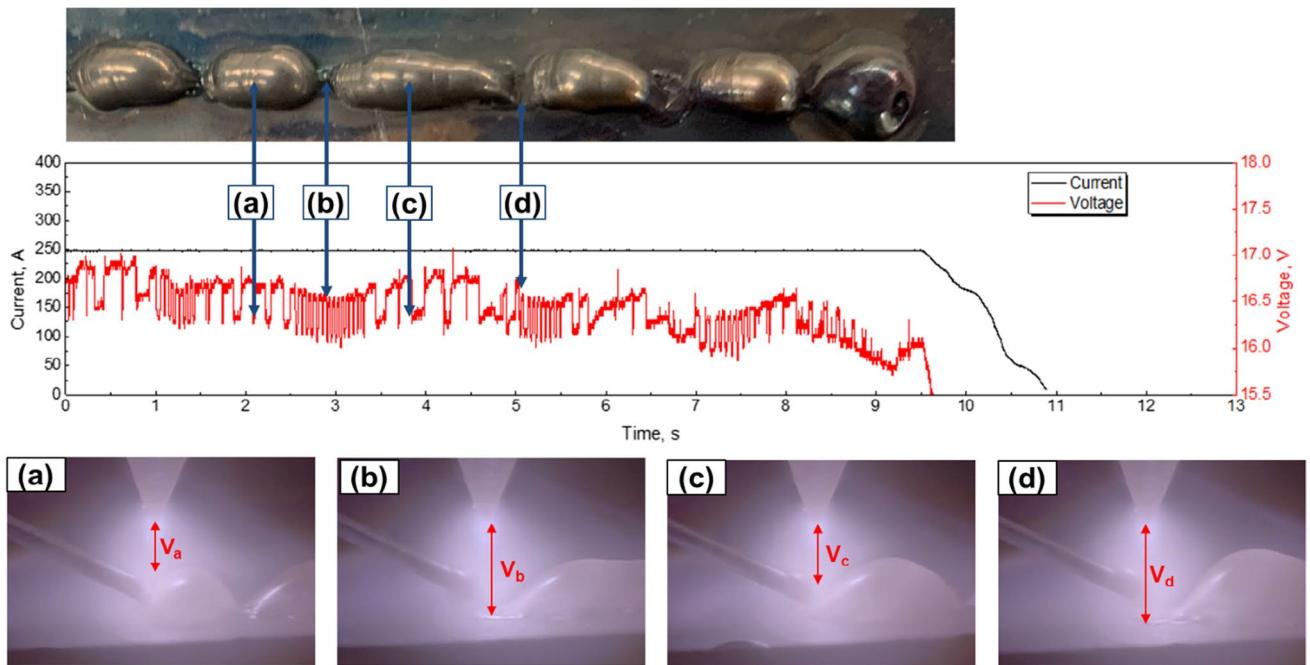
molten pool size and deposition area may decrease when the heat input per unit length of the drop is less than a certain threshold because of low current or high TS [24]. Molten pools are small in size, and a separated and spherical bead area is formed as they gradually move to the rear side. The repetitive occurrence of this phenomenon leads to balling defects.

Figure 3 shows one-dimensional (1D) data of current and voltage and their corresponding bead shapes under the balling condition. The arc length is proportional to the voltage under a constant current, wherein the arc length indicates the distance between the electrode and bead. As shown in Fig. 3a and c, when the bead is formed as balling closer to the electrode, the arc length decreases, leading to a decrease

in voltage. In contrast, a long arc leads to a higher voltage in the separated bead area between two humped beads, as shown in Fig. 3b and d. Hence, voltage profiles are decisive and should be analyzed to detect balling defects as they reflect bead formation.

## 2.2 Anomaly Detection Using Machine Learning

Anomaly detection relies on sensor-data-driven algorithms as they can empirically reflect melt pool behaviors, kinematics, and thermodynamics. Sensor data include heat transfer tracking, surface optical or thermal imaging, melt pool imaging, and melt pool dynamics, providing key information concerning anomaly detection.



**Fig. 3** Balling beads and a voltage profile: short arcs in (a) and (c) and long arcs in (b) and (d)

Accordingly, the relevant literature highly depends on machine learning, which can be further divided into real-time and non-real-time approaches. Non-real-time anomaly detection is used when an anomaly is detected in an ex-situ procedure. Artificial neural networks (ANNs) are commonly used in numerical prediction using sensor data. In contrast, convolutional neural networks (CNNs) are popular for two-dimensional (2D) data, including melt pool images, infrared images, and computed tomography scans. The excellent performance of CNNs was demonstrated as an image detector and classifier specific to anomaly detection problems in industries [9]. A typical CNN structure comprises convolution, pooling, and fully connected layers. The convolution layer includes a set of convolutional kernels for dividing an image into small slices to extract feature motifs and convolving with the image using weights, i.e., by multiplying image tensors with their corresponding slices. The pooling layer reduces input dimensionality and provides spatial invariance to the network. The fully connected layer considers inputs from the preceding layers and derives the outputs of all the layers [25]. CNNs can extract representative features without prior knowledge and reduce the training time by decreasing the weight dimension [26]. However, CNNs require a large amount of training data and heavy computation as the network becomes deeper [27].

Biranchi et al. (2015) suggested a machine learned approach that predicted compressive strength using multi-gene genetic programming and general regression neural network in fused deposition modeling [28]. Scime and Beuth (2018) developed a multi-scale CNN method for detecting diverse defects from image patches in laser powder bed fusion [17]. This method paved the way for in-process defect rectification when a feedback control system was implemented. Jin et al. (2019) derived a CNN encoder and decoder for detecting outliers based on a learned distribution of normal behaviors [12]. Mojahed Yazdi et al. (2020) proposed a deep-learning method for detecting porosity in the internal layers of a cylindrical part [29]. They merged a CNN with an ANN to extract features from image data to generate statistical features. Lyu and Manoochehri (2021) proposed a CNN model for extracting, analyzing, and classifying in-plane anomalies in fused filament fabrication [30]. These studies derived anomaly detection algorithms with over 90% accuracy; however, they could not reach to real-time decision-making for in-situ quality monitoring and control.

In recent times, real-time anomaly detection has become a subject of interest in the AM community for two reasons. First, real-time image acquisition has become more practical because low-cost and robust machine vision systems have become more available. Second, real-time data analysis has become more feasible owing to advances in computing power. Yan et al. (2022) proposed a decomposition-based

method for real-time anomaly detection based on spatio-temporal data in laser powder bed fusion [31]. This method benefited from a layer-wise production paradigm for gathering information on process quality and stability in real-time. Lee et al. (2021) suggested a CNN-based method for detecting anomalies in WAAM based on real-time monitoring using high-dynamic-range (HDR) camera images [19]. Segura et al. (2021) proposed an online framework for detecting droplet anomalies from video images in inkjet printing [32]. Cho et al. (2022) implemented a MobileNet-based real-time anomaly detection system [33]. In this regard, the application of machine learning represents a promising solution for real-time anomaly detection and provides a basis for real-time quality control.

### 2.3 Anomaly Detection Using Transfer Learning

Typical machine learning requires massive training data to achieve acceptable performance. However, collecting sufficient data is significantly challenging as data acquisition is costly. Nevertheless, learning-driven modeling is vital, and TL is a means for overcoming this problem. The major terms in TL are defined as follows:

- Task  $\mathcal{T}$  denotes learning tasks, e.g., regression, prediction, clustering, and classification.  $\mathcal{T}$  comprises a label space  $\mathcal{Y}$  and predictive function  $f(\cdot)$ .
- Domain  $\mathcal{D}$  denotes different feature spaces or marginal probability distributions caused by disparate contexts in which data are generated.  $\mathcal{D}$  consists of a feature space  $\mathcal{X}$  and probability function  $P(X)$ .
- Source denotes a task  $\mathcal{T}_S$  or a knowledge supplier  $\mathcal{D}_S$ .
- Target denotes a task of interest  $\mathcal{T}_T$  or a preceding knowledge consumer  $\mathcal{D}_T$ .
- Knowledge is a broad term that includes instances, features, parameters, relations, and models and thus acts as a transporter between the source and target.

Figure 4 presents the differences between machine learning, inductive TL and transductive TL. As shown in Fig. 4a, machine learning learns data to derive each model for a task. When  $\mathcal{T}$  performs prediction, machine learning produces a model, i.e.,  $\mathcal{Y} = f(\cdot)$ , from training labeled data samples that correlate inputs and outputs of predictions. When  $\mathcal{T}$  performs another task, machine learning would produce a different model specific to the task from data samples. Figure 4b shows inductive TL, where a predictive model is induced in  $\mathcal{D}_T$  using data in  $\mathcal{D}_S$  and  $\mathcal{T}_S$  when  $\mathcal{T}_S \neq \mathcal{T}_T$ , irrespective of the homogeneity between  $\mathcal{D}_S$  and  $\mathcal{D}_T$ . As conditional probabilities  $P(Y|X)$  can be different across tasks, a few labeled data in  $\mathcal{D}_T$  are required to adjust the transfer of conditional probabilities or the discriminative function from  $\mathcal{T}_S$  to  $\mathcal{T}_T$  [34]. Figure 4c shows transductive TL,

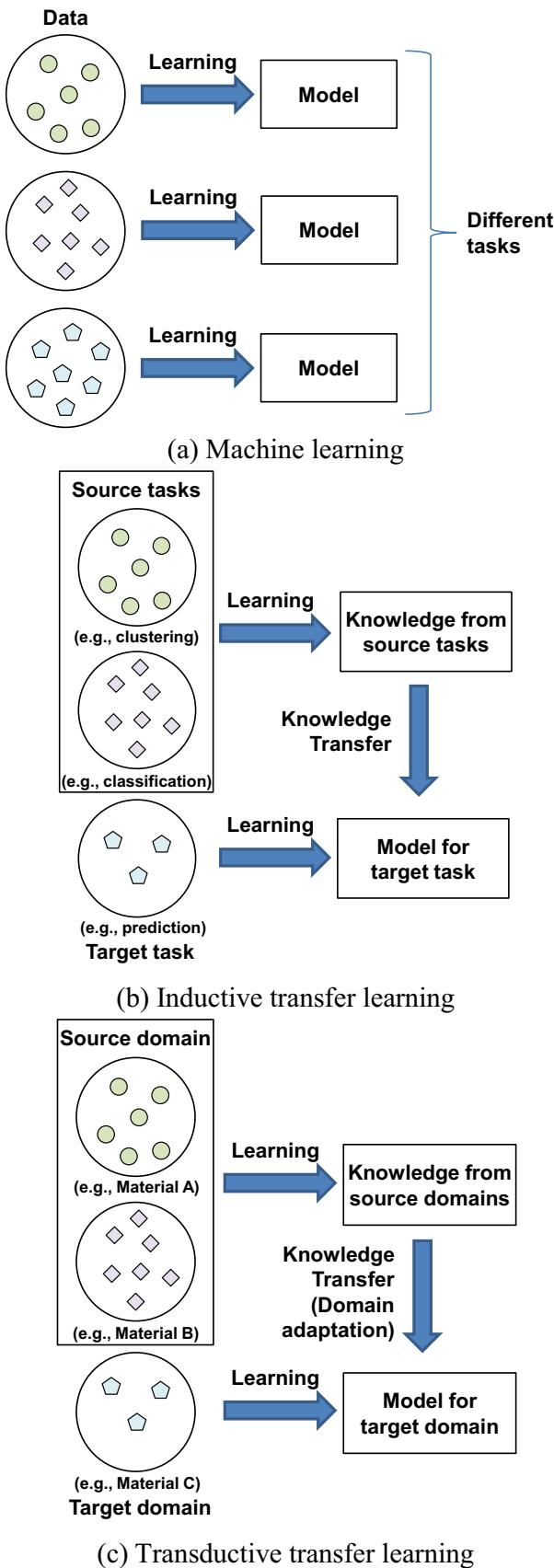


Fig. 4 Machine learning and transfer learning

where a predictive model is transduced in  $\mathcal{D}_T$  when  $\mathcal{T}_S = \mathcal{T}_T$ ; however,  $\mathcal{D}_S \neq \mathcal{D}_T$ . Transductive TL includes two cases: (1) the feature spaces between  $\mathcal{D}_S$  and  $\mathcal{D}_T$  are different, i.e.,  $\mathcal{X}_S \neq \mathcal{X}_T$ , and (2) the feature spaces are the same, i.e.,  $\mathcal{X}_S = \mathcal{X}_T$ ; however, the marginal distributions of the input data are different, i.e.,  $P(X_S) \neq P(X_T)$  [22]. The latter case is identified as domain adaptation, where a difference in the marginal probability distributions existed between the source and target data; thus, the knowledge of the source domains needs to be adapted to the target domain.

TL has been applied to manufacturing to create predictive models for fault diagnostics and anomaly detection [35]. Oquab et al. (2014) suggested a network for training labeled source data and transferring CNN internal layers to a target learner [36]. Shao et al. (2018) employed a deep TL method to diagnose motors, gearboxes, and shaft bearings [37]. Guo et al. (2018) suggested a deep convolutional transfer network for the fault diagnosis of bearings in different machines [38]. Sun et al. (2018) used a sparse autoencoder and deep TL technique to estimate the residual life of a cutting tool [39]. Ferguson et al. (2018) proposed a mask region-based CNN to identify casting defects from X-ray images and perform defect detection and segmentation [40]. Imoto et al. (2018) used a CNN to automate defect classification and the TL network to reduce the labeled data for classifying defects in semiconductor manufacturing [41]. Pan et al. (2019) applied TL to the fault diagnoses of high-voltage circuit breakers [42]. Zellinger et al. (2020) presented a multisource TL method for predicting errors using time-series data for tool settings incorporating domain knowledge [43]. Wang and Gao (2020) proposed a deep learning-based TL model for diagnosing faults in rolling bearings based on vibration analysis [44]. Gong et al. (2020) studied the same concept for detecting defects in aeronautic composite materials using the images of non-destructive X-ray tests [45]. Michau and Fink (2021) proposed an unsupervised TL framework to ensure the alignment of unit distributions for enforcing the conservation of the inherent variability of datasets [46]. Liu et al. (2021) suggested a deep TL approach to extract low-dimensional features for process recognition in milling [47]. Kim et al. (2022) proposed a multisource TL method for creating predictive models of machining power [48]. Marei et al. (2021) applied a TL-enabled CNN approach to estimate the health of cutting tools [49]. Liu et al. (2021) suggested a knowledge reuse strategy for training CNN models to improve defect inspection accuracy for injection molding [50].

Although TL is rarely used for anomaly detection in AM, its application is increasing [51]. Ho et al. (2021) proposed a TL-based method for predicting porosity in real-time using the thermal images of a melt pool [52]. Scime et al. (2020) presented CNN and TL-based models for the pixel-wise semantic segmentation of layer-wise powder-bed image data

[13]. Zhu et al. (2023) developed a TL-based method with applying a parametric and self-supervised object detection model to detect surface morphology in DED [53].

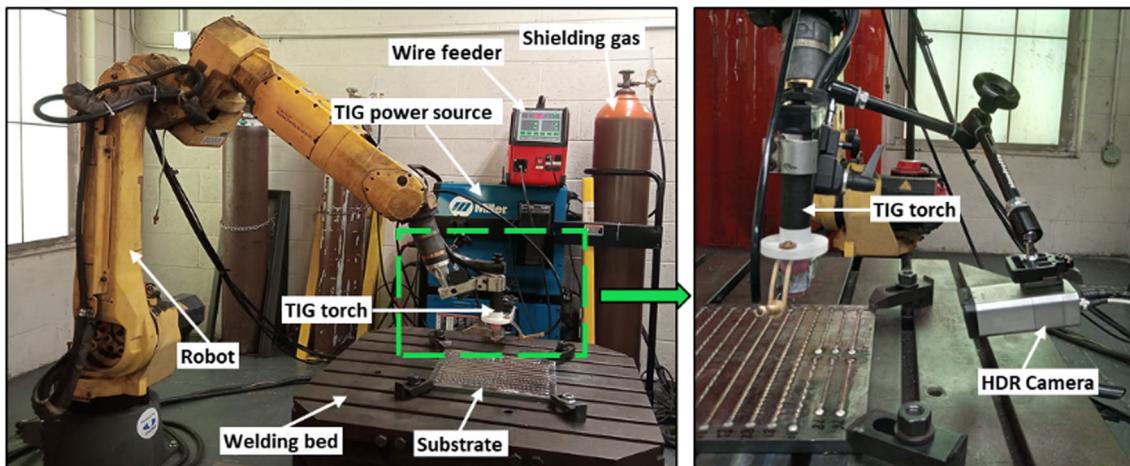
The significance of the present work originates from the need to generate and apply alternative but desirable anomaly detection models in materials where data are insufficient. When anomaly detection models are required for high-cost materials, collecting sufficient data is more challenging. TL is an efficient solution because it adopts a well-trained network even using insufficient data and employs the network across multiple domains. In other words, a low-cost material's (e.g., steel) model can be used to create a high-cost material's (e.g., Ti-6Al-4V) model for anomaly detection. TL enables the network to be trained with a dataset collected from a low-cost material, i.e., source material, the features of which are extracted and stored in the hidden layers inside

the network. These features are adjusted with a high-cost material, i.e., the target material, particularly when the two materials are not distinct in physical and thermal aspects.

### 3 Experiments

#### 3.1 Experimental Setup

The experiments were conducted using a GTAW-based WAAM system, as shown in Fig. 5. Table 1 lists the experimental setup details. The robot arm was moved to the coordinates designated by the controller. As the tungsten inert gas (TIG) torch was attached to the hand of the robot arm and supplied with energy from a source, it deposited a feeding material provided by the wire feeder to generate weld



**Fig. 5** Experimental environment, including wire feeder, shielding gas, TIG power source, TIG torch, robot, and HDR camera

**Table 1** Experimental setup details

System	Module	Name	Description
Gas tungsten arc welding (GTAW)	Robot arm	Fanuc ArcMate 120iC	A six-axis movement for the TIG torch
	Controller	Fanuc R-30iA	Controls the robot arm
	Torch	TIG	Operate arc welding
	Substrate	LCS	70% Argon + 30% Helium shielding gas 5 mm arc length 30 cm × 30 cm × 1.2 cm (size)
	Wire feeder	Generic wire feeder	Feed a wire material
	Power source	Miller Dynasty400	Supply energy source
Numerical data acquisition	Current and voltage sensor	Miller Insight ArcAgent Auto	Measure arc current and arc voltage
	Current and voltage data interface	Miller Insight Centerpoint	Monitor and acquire arc current and voltage data
Image data acquisition	Vision sensor	Weldvis WL2-H7-M35	HDR camera for monitoring welding arc, molten pool, and weld bead images
	Camera data interface	Elgato Game Capture HD	Capture and record image data

beads on the substrate. The TS, WFR, and current were process parameters determined for the controller input, wire feeder, and energy source, respectively. The current and voltage sensors measured the numerical values of arc characteristics in real-time. The data interface monitored and acquired the arc current and voltage data generated by the sensor. An HDR camera was attached to the torch to capture weld pool and bead images along with the movement of the torch. This camera was optimized for arc welding with a dynamic range of 140 dB to capture high-quality video frames. Standard camera systems are inapplicable owing to their low dynamic ranges and lightning interferences in arc welding. The camera data interface recorded the images and converted them into.jpg file formats.

The experiments were designed based on changes in two process parameters: a WFR of 70–300 cm per minute (cpm) and TS of 10–100 cpm with increments of 25 and 10, respectively. The current was maintained at 200 amperes (A). Each pair of parameters generated 100 unique trials per material, as shown in Table 2. Accordingly, 300 trials were executed for three materials: LCS, STS, and INC. Figure 6 shows the samples of bead depositions on a single layer.

### 3.2 Data Acquisition

Three types of data were acquired: (1) bead shapes, (2) numerical voltage data on timestamps, and (3) camera image data on single-layer deposition. The voltage data were measured at a time rate of 1 kHz and stored as.txt files. They formed a profile, as illustrated in Fig. 3. The camera image data were captured at 50 frames per second (fps) and stored as.mp4 files. Each file was partitioned into individual.jpg images at 50 fps. The camera frame image comprised three regions of interest: the metal transfer, arc shape, and weld pool, as shown in Fig. 7. To avoid confusion, the camera image data and voltage image data were defined as follows:

- Camera image data: Melt pool image files (.jpg) obtained from each video file as captured by the HDR camera (Fig. 7).
- Voltage image data: Image files (.jpg) converted and captured based on the time-series numerical voltage data (.txt) as the input data of the models (Fig. 12).

## 4 Method

This study aims to (1) develop a TL-based method for deriving anomaly detection models learned from a single source material and (2) apply the models to detect anomalies in a target material. In this method, the material properties are concatenated between the source and target materials as manufacturing knowledge features in the model to compensate for discrepancies in the melting mechanism. Figure 8 illustrates the overall procedure of the proposed method. Section 3.1 explains the data preprocessing procedures, and Sect. 3.2 introduces the modeling method.

### 4.1 Data Preprocessing

Data preprocessing transforms raw data into high-quality training and testing data. Feature extraction and pattern discovery during learning become complicated when training data include sparse, imprecise, qualitative, faulty, or missing samples [54]. Hence, data preprocessing is essential because model performance mainly relies on the quality and quantity of the training data. Figure 9 shows the data preprocessing procedure, and the subsections hereafter explain the technical details of each step. The input data comprises the deposited bead shapes, numerical voltage data, and camera image data.

**Table 2** Process parameters and bead numbers considered for the experiments

Wire feed rate (cpm)	Travel speed (cpm)									
	10	20	30	40	50	60	70	80	90	100
75	1	11	21	31	41	51	61	71	81	91
100	2	12	22	32	42	52	62	72	82	92
125	3	13	23	33	43	53	63	73	83	93
150	4	14	24	34	44	54	64	74	84	94
175	5	15	25	35	45	55	65	75	85	95
200	6	16	26	36	46	56	66	76	86	96
225	7	17	27	37	47	57	67	77	87	97
250	8	18	28	38	48	58	68	78	88	98
275	9	19	29	39	49	59	69	79	89	99
300	10	20	30	40	50	60	70	80	90	100

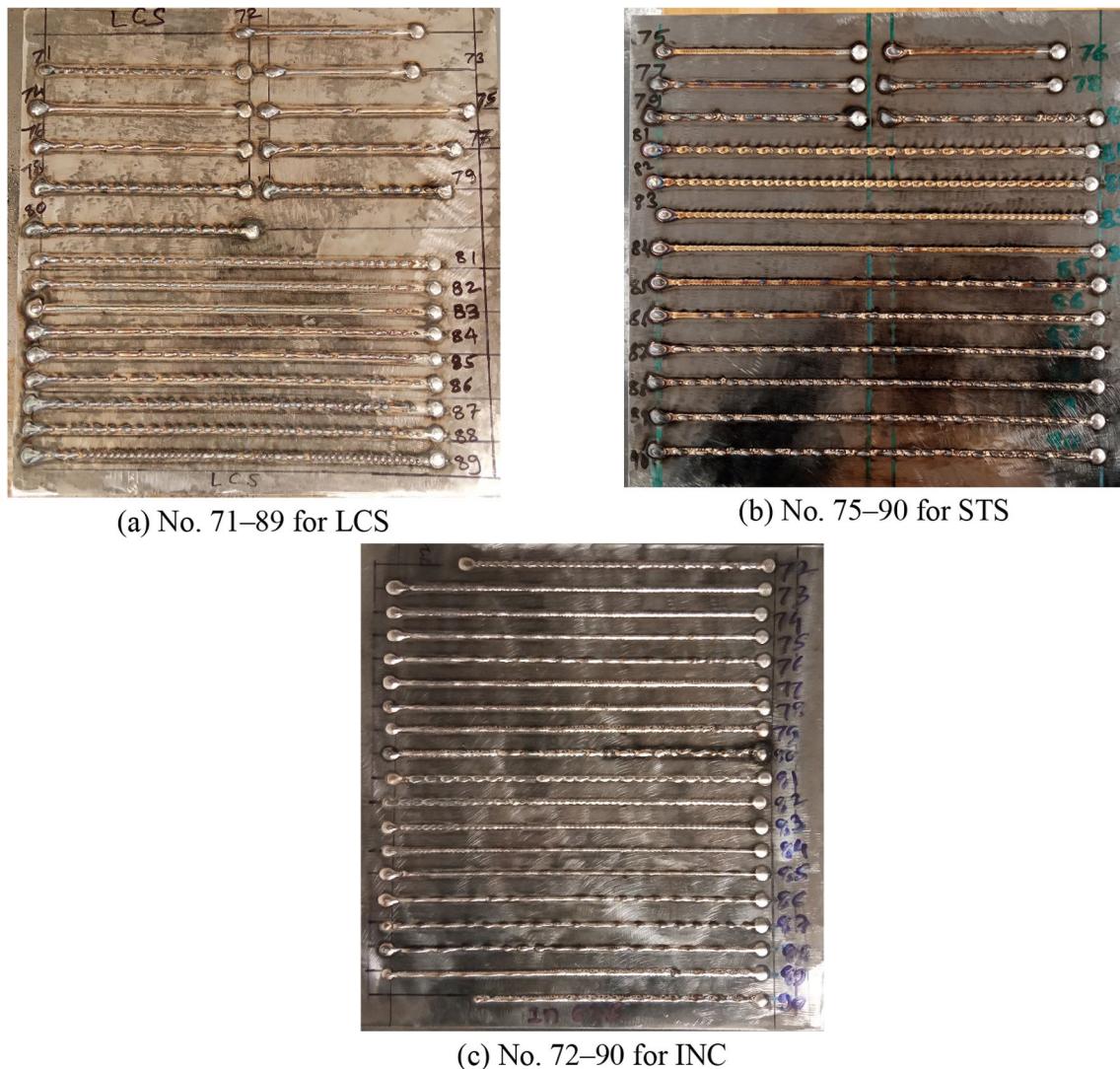


Fig. 6 Bead depositions for experiments

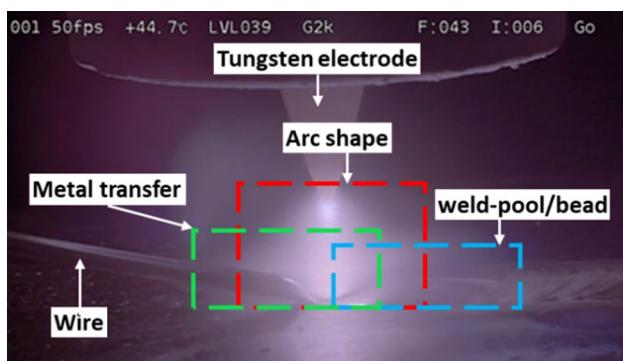


Fig. 7 Camera image data, including metal transfer, arc shape, and weld pool

#### 4.1.1 Bead Classification and Balancing

Bead classification is necessary to distinguish regular data patterns. Beads classifiable as normal and abnormal are considered, while unclassifiable beads due to their shape change or data irregularity are excluded.

First, each bead was classified and labeled as *normal*, *abnormal*, or *unclassified* based on the judgments of two experts with unaided eyes. Figure 10 shows the voltage data profiles of the corresponding bead shapes. As shown in Fig. 10a and b, the two beads are classified as *normal* because they are well-formed with stable and smooth data patterns. As shown in Fig. 10c and d, the beads are classified as *abnormal* because they contain balling defects along the trajectories. As shown in Fig. 10e and f, the beads are classified as *unclassified* owing to their transition from normal to abnormal states and vice versa. Theoretically, a

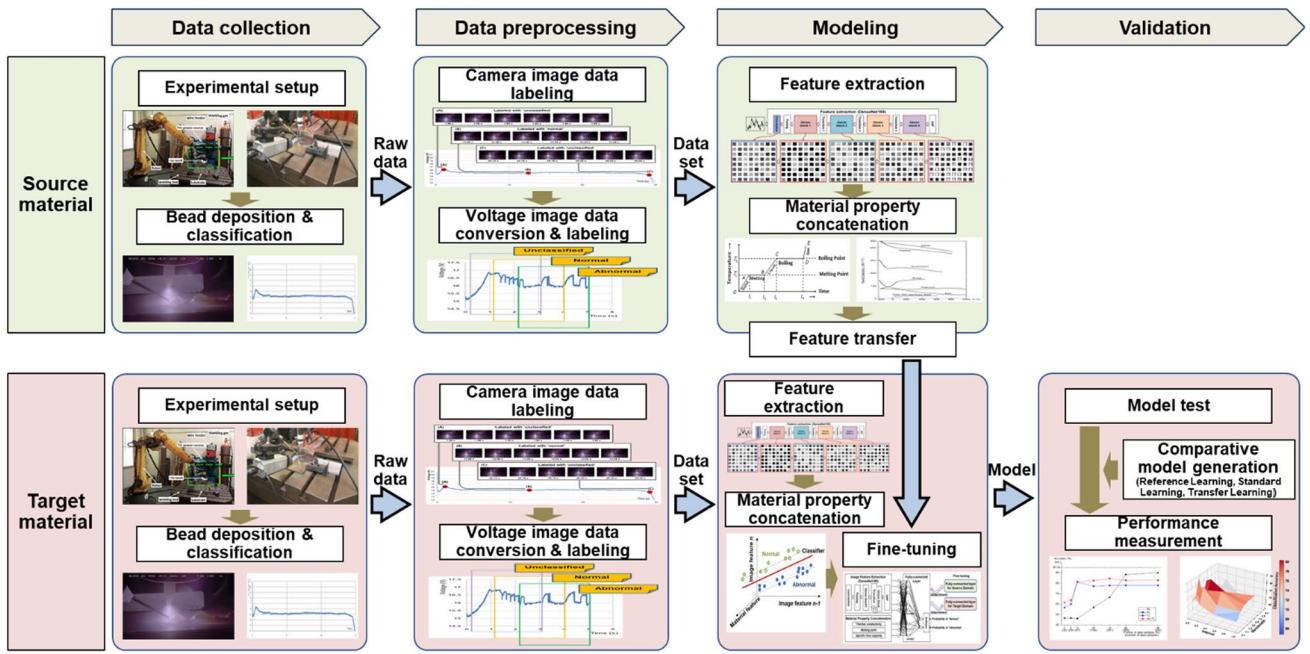


Fig. 8 Procedure of material-concatenated transfer learning

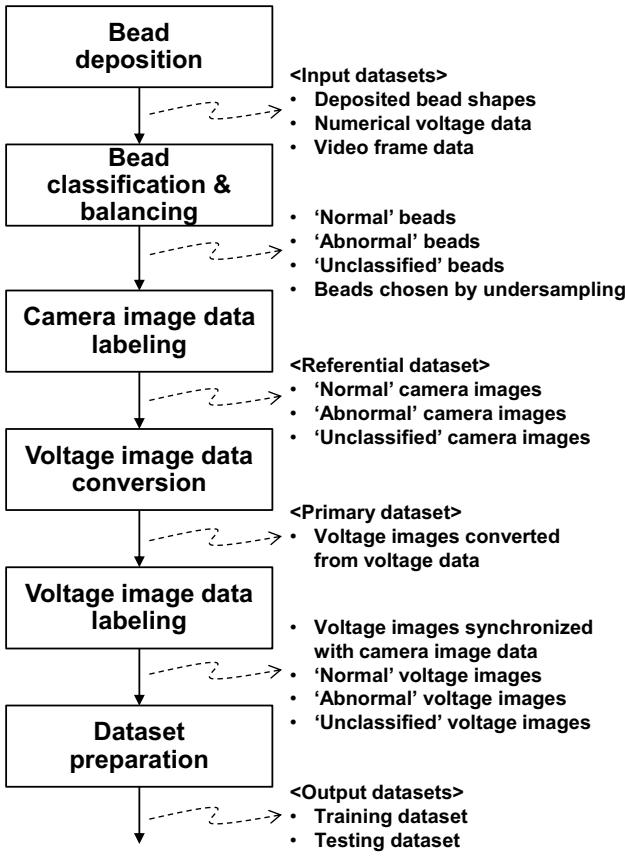


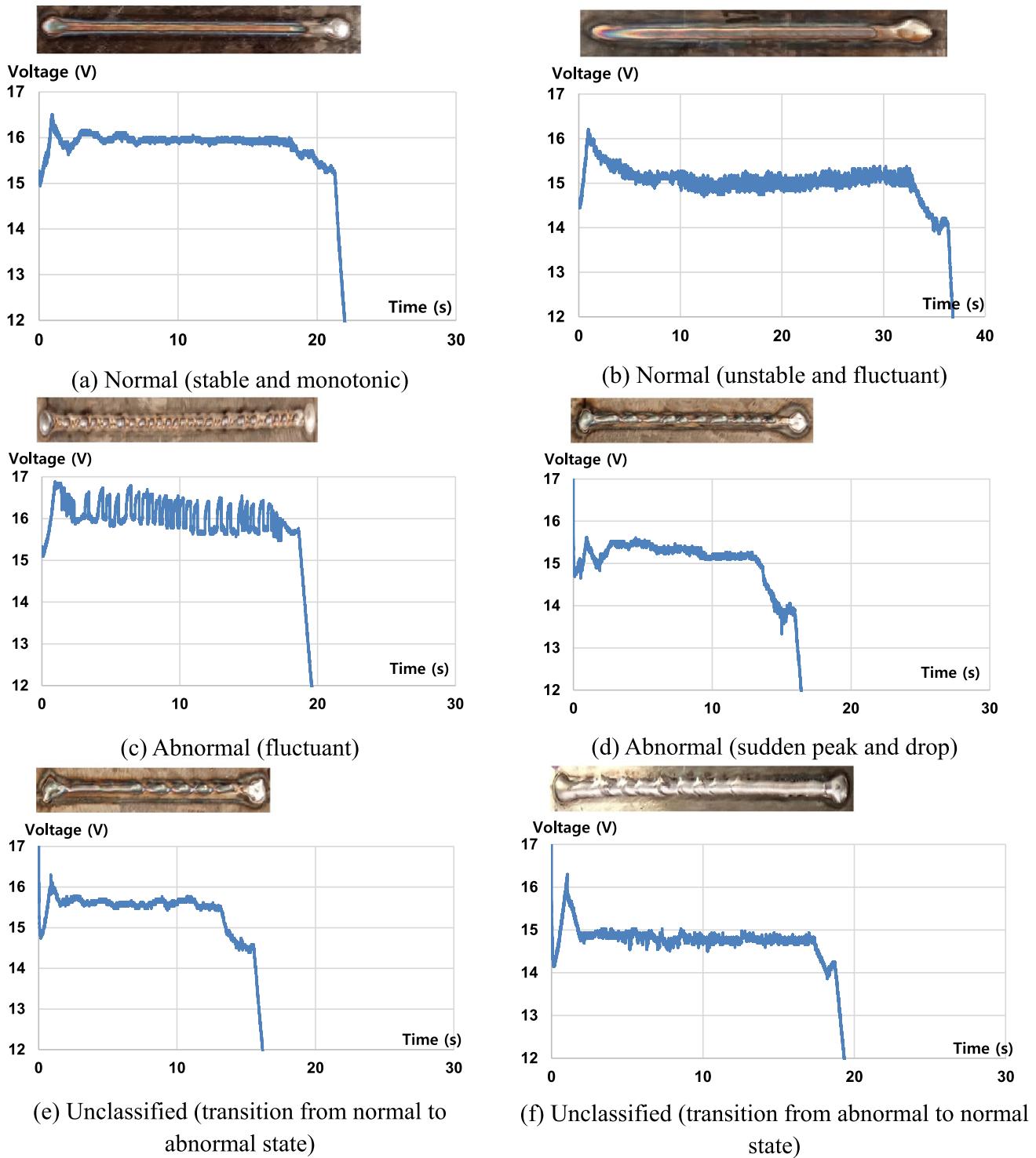
Fig. 9 Data preprocessing procedure

bead trajectory should maintain consistency in shape without a state transition because the process parameters do not change during deposition. However, state transitions can occur in practice owing to external and uncontrolled factors, including deposition instabilities, feeding material irregularities, and unknown reasons. Unintentional state transitions can induce vagueness in pattern separation, thus making it desirable to exclude beads containing such state transitions, as illustrated in Fig. 10e and f.

Second, beads associated with stable and monotonic voltage profiles were considered among all the beads classified as *normal*. The voltage profile in Fig. 10a appears to be more stable than that shown in Fig. 10b, although both are normal, implying that even the same normal beads can possess different voltage patterns because the melting mechanism influences voltage values, as explained in Sect. 2.1. Even in the same normal state, such unstable and fluctuating patterns can decrease accuracy. For instance, the bead shown in Fig. 10a was considered in the classification process, whereas the bead shown in Fig. 10b was excluded.

It is essential to resolve the class imbalance problem while classifying beads. The class imbalance problem is referred as a dataset with a skewed ratio of majority to minority samples, as it frequently occurred in a data-scarce and normal-biased environment [27]. This problem should be resolved because it can cause overfitting, resulting in a small learning error during training but a high prediction error during testing.

The class imbalance problem was also observed in this study because more normal data were generated than

**Fig. 10** Patterns of voltage data profiles

abnormal data. It is unable to learn data patterns correctly for the abnormal state if the training data are extremely biased toward the normal state. Accordingly, balancing the numbers of normal and abnormal samples is a solution to this problem. Oversampling and undersampling can

be used to address this issue. The former increases the size of the minority class to balance the majority class, whereas the latter reduces the size of the majority class to balance the minority class [55]. In this study, undersampling was used to resolve the class imbalance problem.

In undersampling, the amount of normal data is reduced by excluding normal beads to achieve a desirable balance with the amount of abnormal data. This procedure was performed by considering the balance of the total time consumed to fabricate the beads. Table 3 lists the bead numbers selected and classified as *normal* or *abnormal*. Each value in brackets represents the fabrication time (s). The type indicates whether a bead is used for a training or testing dataset.

#### 4.1.2 Camera Image Data Labeling

Camera image data labeling was performed by tagging each camera image frame with a classifier consisting of *normal*, *abnormal*, or *unclassified*. This labeling was intended to assign the same classifier to voltage image data with the corresponding camera image data along with timestamps. The two experts also analyzed the individual frames with unaided eyes and manually labeled each classifier.

This labeling was performed easily because the bead classification was already completed, as described in Sect. 4.1.1. The camera image data belonging to a normal or abnormal bead were labeled as *normal* or *abnormal*, respectively. However, some camera image data were labeled as *unclassified* when they belonged to the starting and ending spots on the bead trajectory because TS was zero at both ends, where the torch did not move for a short time. Figure 11 shows camera image data labeling at bead No. 25 in INC. The camera image data at spots (a) and (c) are labeled *unclassified*. Meanwhile, the camera image data for the ordinary period (b) are labeled normal because the bead was classified as *normal*. If a bead was classified

as *abnormal*, the camera image data would be abnormal during the ordinary period.

#### 4.1.3 Voltage Image Data Conversion

The time-series voltage data were converted into voltage image data in the time domain to apply image-based learning for time-series classification. The time-series data are originally analyzed using three approaches: (1) The model-based method generated an underlying model using Markov and statistical models, (2) The distance-based method measured the similarity between two sets of time series using a distance function, and (3) The feature-based method used Fourier and discrete wavelet transforms to transform the time series into a set of representing features [56]. However, these approaches have drawbacks in practice because (1) the time-series data must satisfy the stationary assumption, (2) the length of the two sets of time series must be equal to high sensitivity, and (3) feature selection cannot be easily performed without discretization owing to information loss. These drawbacks can adversely impact data quality. In this context, applying image-based learning to time-series data can become viable for time-series classification [57]. This enables to extract features automatically without prior knowledge and handle noisy data properly by discarding them at each subsequent layer [58].

For image conversion, the size and frequency of each voltage image were required to be determined to segregate the voltage data profile into a series of image snapshots. The bandwidth was used for the image size, and the interval was employed for the image frequency. These are defined as follows:

**Table 3** Selected beads and their corresponding fabrication times

Material	Normal bead		Abnormal bead		Type
	Bead no. (time (s))	Total time (s)	Bead no. (time)	Total time (s)	
LCS	5(75.7), 13(42.3), 14(39.6), 32(24.3), 33(25.2), 34(24.5)	231.6	50(16.9), 68(16.8), 69(17.1), 79(12.1), 80(15.3), 81(20.0), 86(20.2), 88(19.6), 89(20.6), 90(21.6), 91(19.7), 93(18.4), 95(20.3), 96(20.0), 97(19.4), 98(21.2), 99(20.1)	319.3	Training
STS	22(30.2)	30.2	78(14.0), 87(20.8)	34.8	Testing
INC	9(78.4), 17(37.4), 28(27.8), 30(27.9), 36(22.4), 48(19.3)	213.2	61(14.3), 81(22.0), 82(21.7), 88(22.0), 89(22.4), 90(22.6), 92(20.9), 93(20.9), 97(20.0), 98(19.1), 100(19.7)	225.6	Training
	27(26.5)	26.5	51(16.2), 91(16.5)	32.7	Testing
	6(79.8), 16(42.3), 24(30.7), 33(24.5), 44(20.5), 56(17.4)	215.2	30(30.6), 41(21.1), 59(17.9), 61(14.8), 71(14.5), 76(21.9), 79(21.7), 81(20.4), 82(19.2), 83(19.9), 86(20.9), 88(20.7), 91(18.7), 99(17.2), 100(18.2)	297.7	Training
	25(30.2)	30.2	72(18.9), 87(19.3)	38.2	Testing

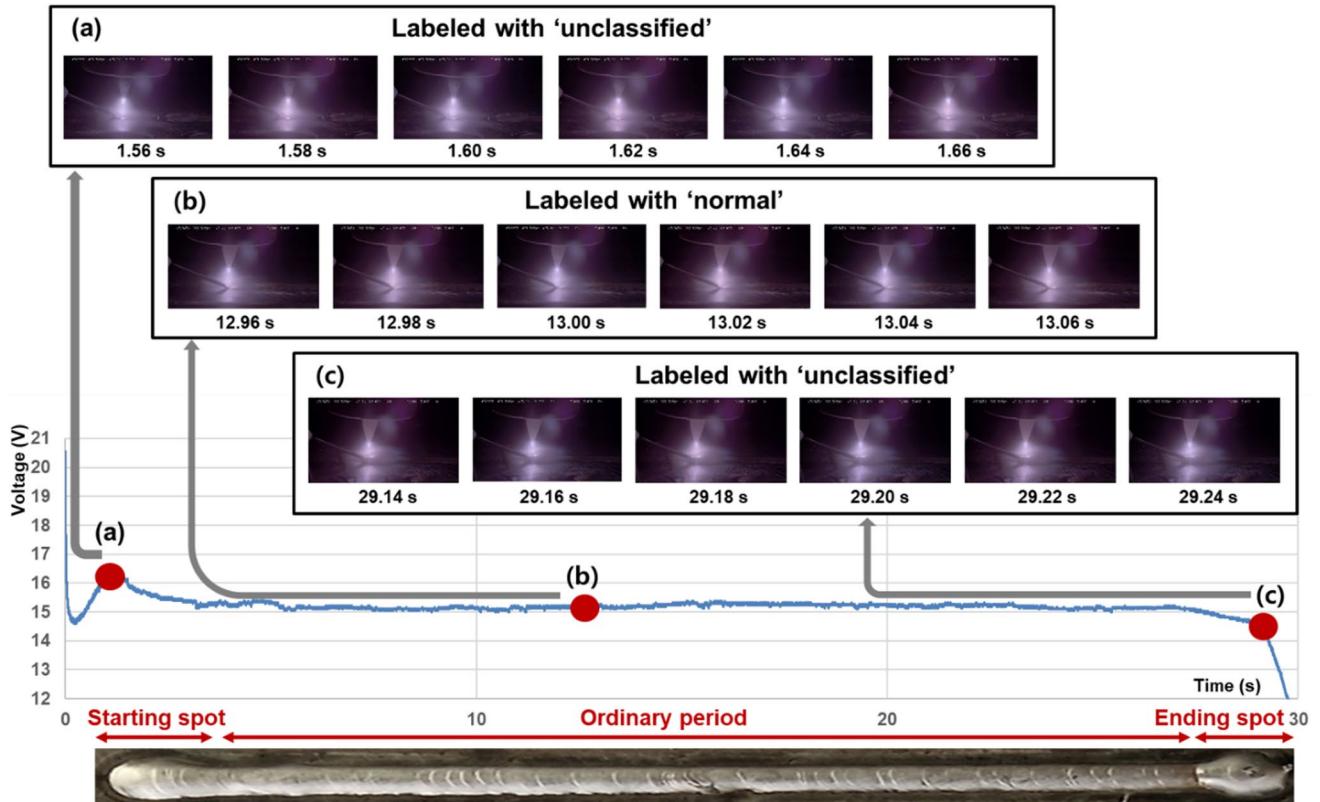


Fig. 11 Camera image data labeling, including **a** labeled with ‘unclassified’, **b** labeled with ‘normal’, and **c** labeled with ‘unclassified’

- Bandwidth is the duration from the earliest to the latest time point on an image snapshot. Bandwidth determines the number of data points in the snapshot of a partial voltage profile.
- Interval is the time gap between the current and next image snapshots. The interval determines the number of snapshots to be converted from the voltage profile.

Equations (1) and (2) express the  $k$ -th set of time points ( $M_{w,i,k}$ ) and their corresponding voltage values ( $V_{w,i,k}$ ). Figure 12 shows the concepts of bandwidths and intervals, where  $w=3$ ,  $i=1$ , and  $k=1, 2$ , or  $3$ .  $V_{3,1,1}$  includes a set of voltage values, i.e., 3000, from the first ( $t=1/1000$  s) to the last ( $t=3000/1000$  s) time points.  $V_{3,1,2}$  includes another set of voltage values (3000) from the first ( $t=1001/1000$ ) to the last ( $t=4000/1000$ ) time points. In other words, each snapshot possesses a 3 s bandwidth; accordingly, each snapshot includes 3000 voltage values because the measurement cycle is set to 1 kHz. The snapshots were periodically generated at intervals of 1 s, along with the voltage profile. Thus, the voltage image data were generated based on the designated bandwidth and intervals.

$$M_{w,i,k} = \{x + ji(k-1) | x \in \mathbb{N}, 1 \leq x \leq jw\}, \quad (1)$$

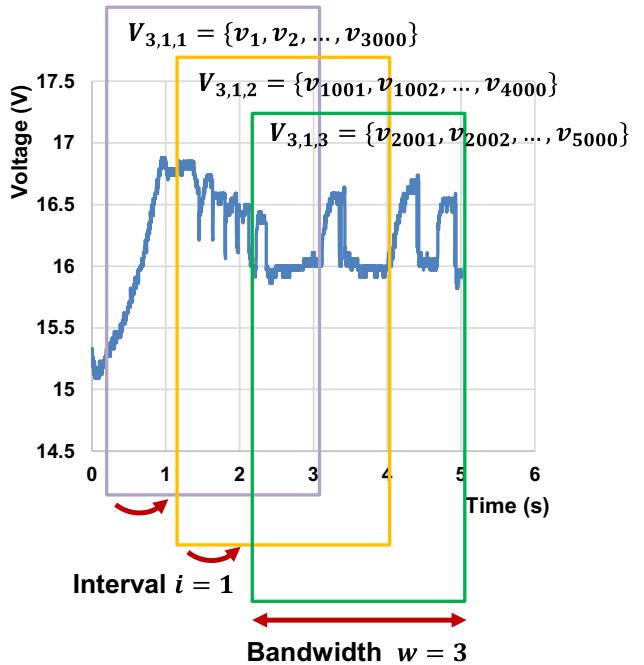


Fig. 12 Bandwidth and interval

where  $w$  denotes the bandwidth,  $i$  denotes the interval,  $x$  denotes the time point,  $j$  denotes the measurement cycle ( $j=1,000$  as a constant), and  $k$  denotes the  $k$ th snapshot.

$$V_{w,i,k} = \{v_m | m \in M_{w,i,k}\}, \quad (2)$$

where  $v_m$  denotes the voltage value observed at time  $(t)=m/j$ .

The bandwidth and interval need to be rationally determined because they directly affect accuracy and training time during model training. If the interval is extremely short, the snapshots can substantially overlap at the current and following time points, thereby sharing a large number of the same  $v_m$ . Data overlapping may disturb feature extraction during training owing to feature similarity. In contrast, features may be sparsely extracted if the interval is extremely long, thus deteriorating the accuracy. Similarly, a large bandwidth can positively affect the accuracy by providing sufficient data samples; however, it may adversely impact the accuracy owing to feature similarity caused by data overlapping. Thus, the relationship between the bandwidth and interval was analyzed using accuracy and training time because they had to be decided heuristically. In this study, the bandwidth was determined as 2 ( $w=2$ ), and the interval was determined as 0.1 ( $i=0.1$ ), because these two values exhibit the best accuracy in the sensitivity analysis.

Then, the height of each snapshot was adjusted to maintain the same image size, as the same-size images need to be used to correctly extract representative features. The image sizes varied depending on the minimum and maximum values of  $v_m$  because they ranged from 0 to 21 V. Therefore,  $224 \times 224$  pixels were preserved in every image owing to automatic adjustment, and the image height was adaptively set using  $v_{range}$  ( $=v_{max}-v_{min}$ ). Figure 13 shows the mechanism of automatic adjustment. The first and second snapshots exhibited larger  $v_{range}$  than the third snapshot because the two formers demonstrated higher voltage values than those in the latter. If  $v_{range,k}=v_{range,k-1}$ , the image features cannot be well extracted during the period with slight fluctuations because the large value of  $v_{range,k-1}$  dominates the determination of the horizontal pixel size. As shown in the first two

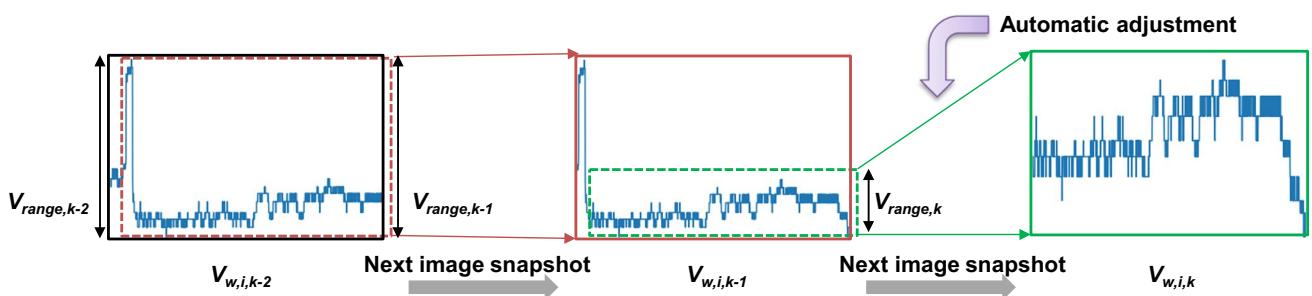
images, the image heights are automatically adjusted with a slight difference, whereas  $v_{range,k}$  was adjusted considering a large extension to focus on the slightly fluctuating period in the third image.

#### 4.1.4 Voltage Image Data Labeling

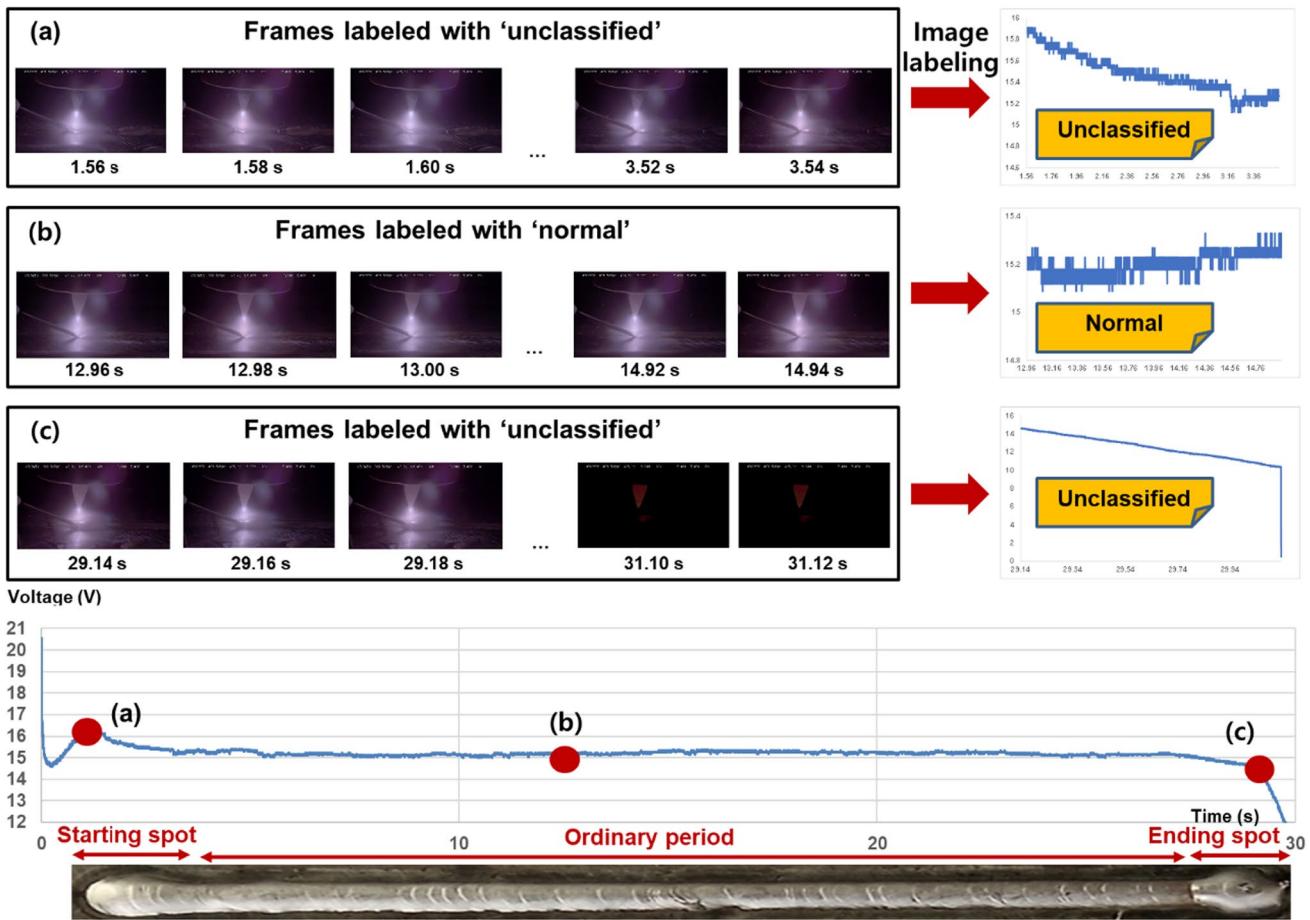
The individual voltage image data were labeled *normal*, *abnormal*, and *unclassified*. All the voltage image data were synchronized with their camera image data with regard to the timestamps. This time synchronization enabled easy labeling using the camera image data labels as a reference. As shown in Fig. 14, the images in the ordinary period (b) were labeled *normal* when their corresponding frames were labeled *normal*. The voltage image data were labeled *unclassified* when they belonged to the starting (a) and ending (c) spots. The unclassified voltage image data were excluded from the training and testing datasets. If camera image data were labeled *abnormal*, their corresponding voltage image data would be labeled *abnormal*. Thus, voltage image data classified as either *normal* or *abnormal* were obtained.

#### 4.1.5 Dataset Preparation

The training and testing datasets were prepared using voltage image data labeled as *normal* or *abnormal*. Beads were randomly selected to construct testing datasets for each material. The classifiers labeled in the testing voltage image data were used only for validation purposes. Table 4 lists the data samples for the training and testing datasets for the three materials. A common 70:30 ratio was used to segregate each training dataset into a training or validation dataset. The training datasets were used for model training, whereas the validation datasets were used for measuring learning errors, which indicated the accuracy of the models during training. The testing datasets were used to measure prediction errors, which represented the accuracy of the learned models in predicting from future data.



**Fig. 13** Automatic adjustment (at bandwidth = 2 s and interval = 0.1 s)



**Fig. 14** Image labeling procedure, including **a** frames labeled with ‘unclassified’, **b** frames labeled with ‘normal’ or ‘abnormal’, **c** frames labeled with ‘unclassified’

**Table 4** Numbers of data samples

Material	Training						Testing		
	Training			Validation			Normal	Abnormal	Total
	Normal	Abnormal	Total	Normal	Abnormal	Total			
LCS	1224	1385	2609	566	553	1119	2115	1420	3535
STS	1093	844	1937	475	356	831	1328	1332	2660
INC	1128	1379	2507	522	553	1075	1476	1644	3120
Total	3445	3608	7053	1563	1462	3025	4919	4396	9315

## 4.2 Modeling

A TL-based method was developed to derive anomaly detection models trained in the source material and apply them to the target material. Figure 15 illustrates the model architecture structurally separated into the source and target domains. In this study, CNNs are used as image feature extractors in the source and target domains. It is because CNNs can identify and extract 2D image features accurately from the voltage image data that were formed into

line-typed and waveformd signals. Additionally, CNNs allow to add different-types of features on their original image features and thus provide flexibility to concatenate material properties. Furthermore, our prior studies, including Cho et al. (2022) [33] and Kim et al. (2023) [59], demonstrated the performance advantage of CNNs in anomaly detection problems, compared with You Only Look Once (YOLO), which is also known as a good object identifier and classifier for real-time applications.

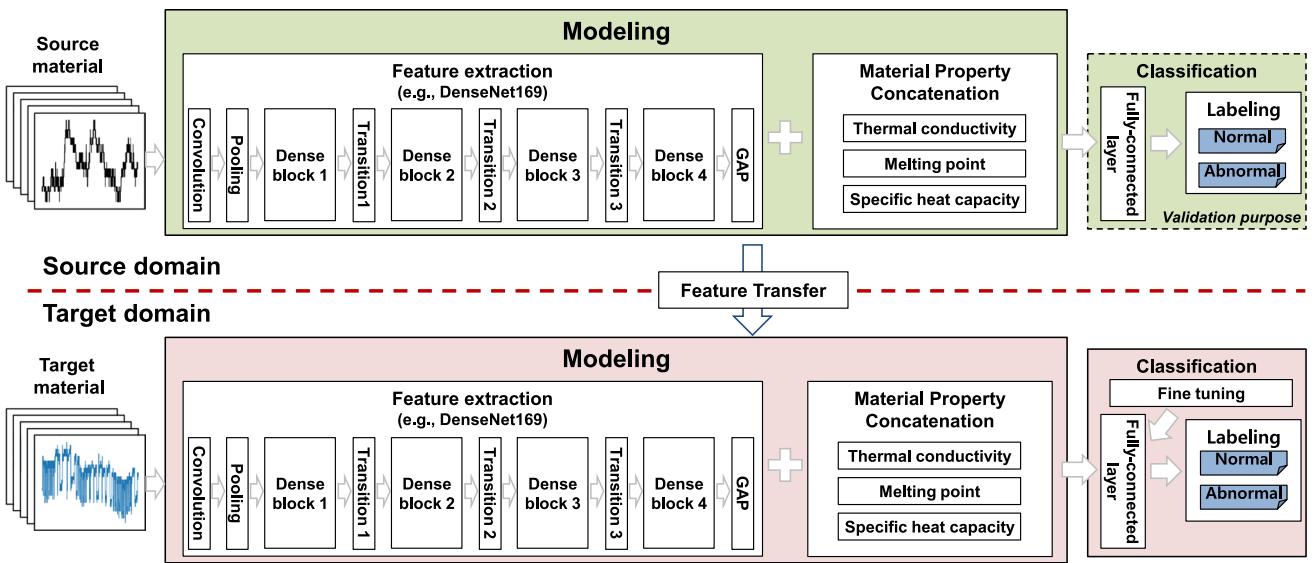


Fig. 15 Model architecture

The proposed method belongs to transductive TL because the feature spaces are the same; however, the marginal distributions of the input data are different. Typical transductive TL would perform domain adaptation by transferring the extracted image features to the target material domain. In contrast, the proposed method performed domain adaptation by transferring the extracted image features and concatenating material properties (particularly thermal properties) as manufacturing knowledge features to accommodate melting characteristics. The features extracted from the voltage image data and the features concatenated from thermal properties were transferred to the target material domain. In the target domain, feature extraction and property concatenation were performed using the data involved in the target material to learn and extract image features and concatenate thermal properties, similar to those in the source domain. Anomaly detection models were developed, and classification was performed to classify each image as *normal* or *abnormal* for use in the target material. In particular, fine-tuning was designed to calibrate weights suitable for the target material to enhance accuracy.

#### 4.2.1 Feature Extraction

Features were extracted from the voltage image data using CNN techniques. A good feature extractor must be selected among various CNN techniques to precisely extract representative image features. A preliminary investigation was performed to select the best feature extractor among four candidates: DenseNet169, InceptionV3, ResNet101, and Xception. These extractors were implemented using the Keras TensorFlow library in Python. TensorFlow is an

open-source framework for machine learning and the Keras library is an open-source machine learning library providing neural network application programming interfaces. The hyperparameters were Adam as an optimizer, categorical cross-entropy as a loss function, 30 epochs, and a batch size of 16.

The best feature extractor was selected by evaluating accuracies of the models derived by TL without material property concatenation (typical TL), and TL with material property concatenation (the proposed method). The training and testing datasets listed in Table 4 were identically used for this feature extractor selection. Models were generated for LCS (source material) using its full set of training (2609) and validation (1119) data samples. These models were applied for the typical or proposed TL and then evaluated for INC (target material) using its full set of testing (3120) data samples. Figure 16 shows the accuracy (%) of the two models. The DenseNet169 model was selected as the feature extractor because it exhibits the best accuracy (84.28% and 89.38%) for both cases.

Figure 17 illustrates the extracted image feature maps and layers. DenseNet automatically extracts representative features by passing image data through convolutional layers [60]. As it connects all layers in a feed-forward manner, the feature maps of the preceding layers act as inputs to the subsequent layer. The dense block concatenates the features of the preceding layers instead of adding them, thus differentiating between the information added to the network and the information preserved [25]. The transition layer consists of a batch normalization layer, rectified linear units (ReLU), i.e., a  $1 \times 1$  convolutional layer, and a  $2 \times 2$  average pooling layer, and performs down-sampling

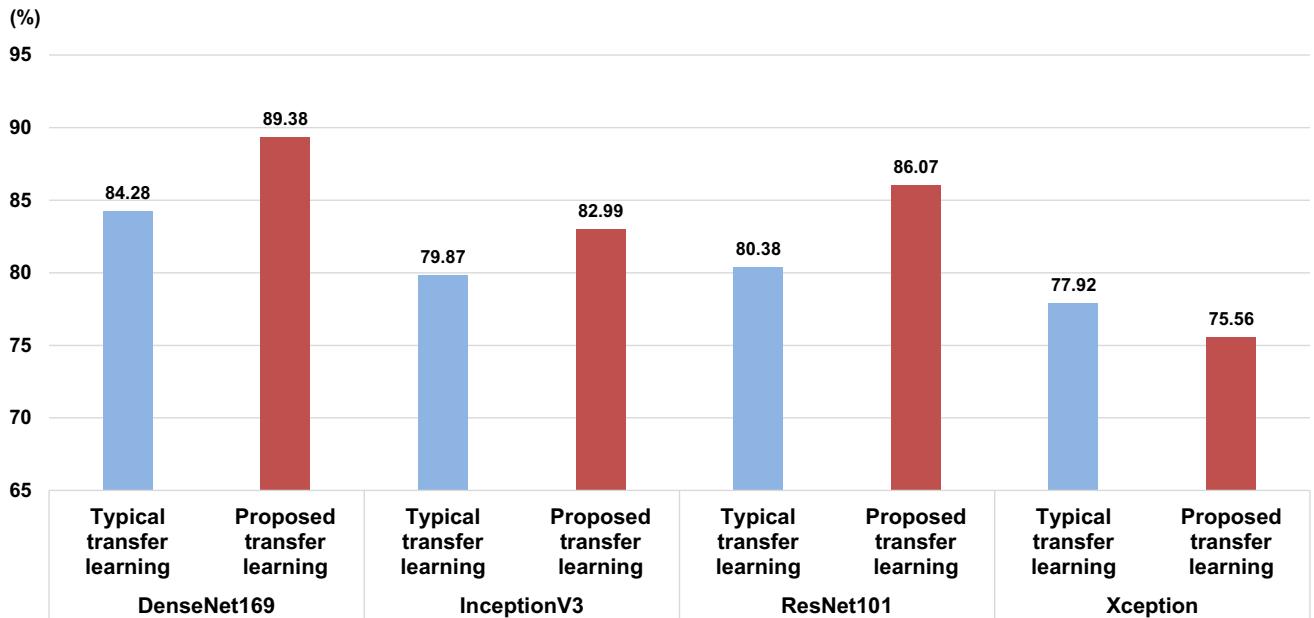


Fig. 16 Preliminary comparison of accuracies

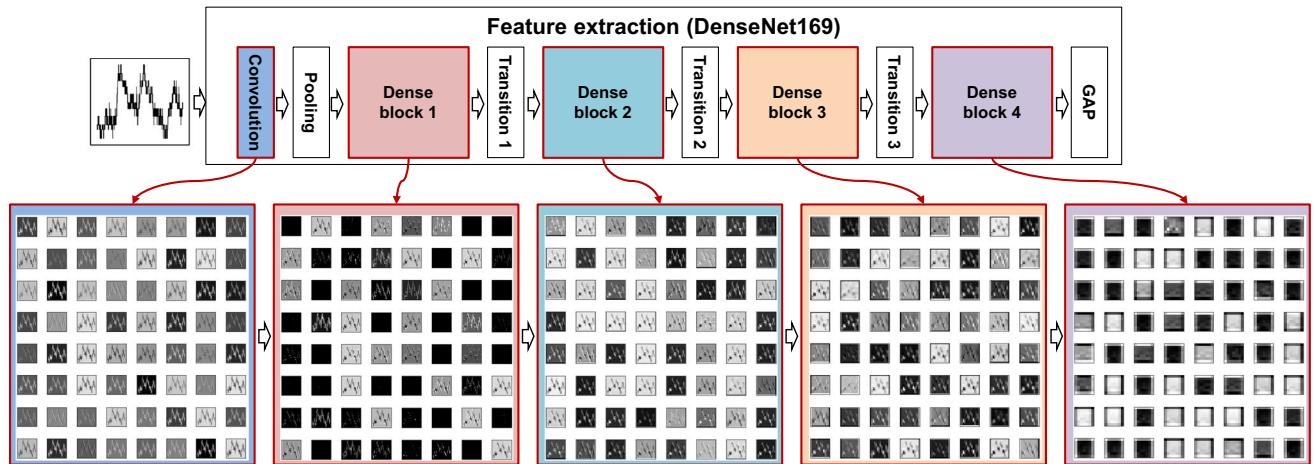


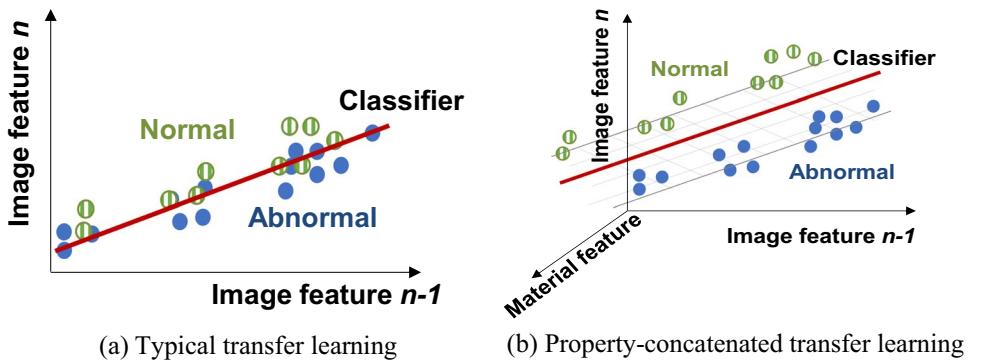
Fig. 17 Feature maps in DenseNet169

to change the sizes of feature maps [61]. The convolutional layer comprises convolutional kernels, where each neuron acts as a kernel. These kernels divide an image into small slices to extract feature motifs, known as receptive fields, and convolve with the image using weights by multiplying the image tensors with the elements of the receptive field [25]. The receptive field captures more global cues than local cues as it increases along the feature hierarchy [62]. As shown in Fig. 17, the feature maps become pixel-wise, abstract, and complicated while they are evolutionarily trained to capture global cues throughout layers.

#### 4.2.2 Material Property Concatenation

Typical TL is limited to explore the distributional difference between the source and target domains, particularly when image features in two different domains exhibit slight deviations within consistent distributions, as shown in Fig. 18a. This phenomenon is called a distributional equality problem. This problem frequently occurs in machine learning and needs to be resolved because it could deteriorate the model accuracy. In this study, this problem was addressed using material property features. Figure 18b shows that

**Fig. 18** Classification dimensions

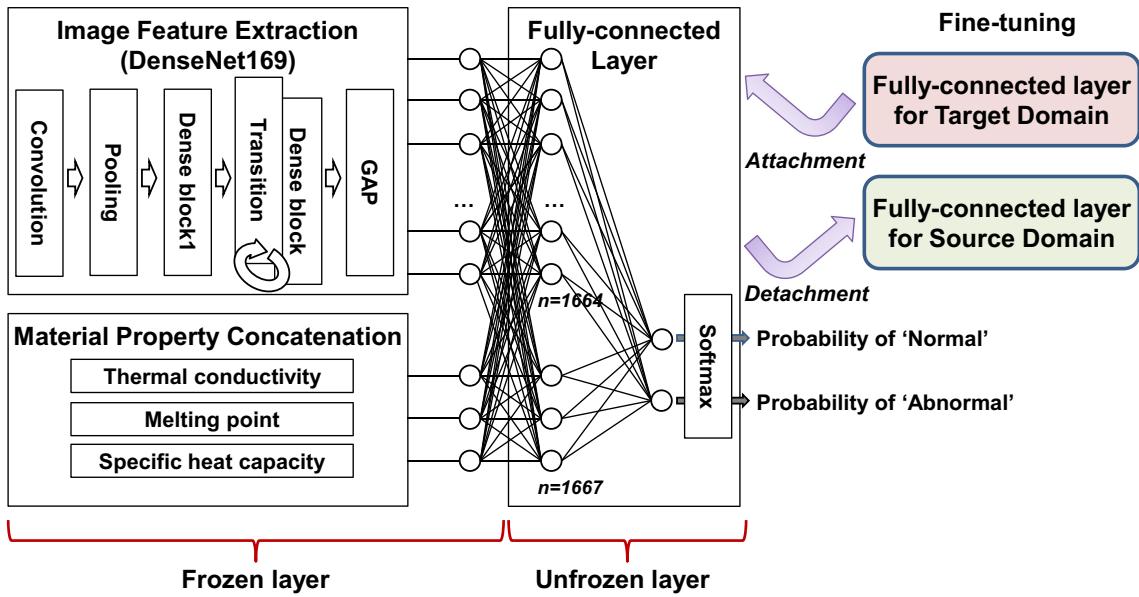


**Table 5** Thermal properties of feeding materials

Property	Unit	LCS	STS	INC
Thermal conductivity	W/m K	42.5	16.3	9.8
Melting point	°C	1430	1370	1290
Specific heat capacity	J/g °C	0.47	0.5	0.41

the material property features were concatenated with the image features. This material concatenation aims to make a small deviation to a large deviation in the image feature distributions between the source and target domains. The domain adaptation originates from expert knowledge, which is related to the influence of material properties on melting and hardening characteristics in WAAM. This knowledge in that certain materials tends to be vulnerable to specific defects owing to their distinctive thermal deformations [8].

Each material exhibits unique physical, chemical, thermal, mechanical, and electrical properties. Among them, thermal properties affect melting and hardening characteristics in WAAM. Hence, thermal conductivity, melting point, and specific heat capacity were identified as material property features in this study. This identification comes from that they are primary thermal properties. Table 5 lists the thermal property values of LCS, STS, and INC; these values were obtained from a material database provided by MatWeb [63]. As these values possessed different units and ranges, normalization was performed to scale and rearrange the original property values into specific values. Min–max (0–1) normalization was applied while concatenating the material property features into modeling.



**Fig. 19** Classification and fine-tuning procedure

#### 4.2.3 Classification

Classification involved labeling each image as *normal* or *abnormal* in the target domain. As shown in Fig. 19, the fully connected layer globally analyzes the outputs of all the preceding layers subordinated in the modeling stage. This layer classifies each image by constructing a nonlinear combination of the selected features and uses common classifiers in machine learning, e.g., support vector machines, softmax, and ANNs. Classification necessitates fine-tuning, which represents the training of new data based on a set of pre-trained weights [64]. Fine-tuning is essential for performance improvement in TL because it applies pre-trained models acquired from the source domain to the target domain.

TL uses a CNN-based model pre-trained on image data. The proposed method was built upon a CNN model pre-trained from voltage image data and property concatenation. This method considers all image and material-property features as a whole set of features at the front layers in a CNN model. However, it trains only in the last layer, using the data from the target domain. This method enables feature extraction from a large amount of voltage image data in the source domain; however, fine-tuning uses a small amount of voltage image data in the target domain. Thus, classification facilitates labeling an image using a *normal* or *abnormal* classifier.

In Fig. 19, fine-tuning preserved the weights of the pre-trained CNN model in some layers and tuned them in others. The front layers are frozen to preserve their weights, as features are obtained from these layers. In contrast, the last layer becomes an unfrozen layer to revise the weights to accommodate features specific to the target data [65]. The preceding layer constitutes 1664 output nodes from the global average pooling (GAP) for image feature extraction and three output nodes from material property concatenation. The unfrozen layer uses 1667 nodes connected to the frozen layer as input. This layer uses the softmax activation function and comprises two output nodes for classifying a data sample as normal or abnormal. This binary classification is decided based on a higher probability of normal or abnormal occurrence, wherein the sum of both probabilities equals 1.

### 5 Validation and Discussion

#### 5.1 Validation

The proposed method was validated in terms of accuracy using a testing dataset. The computing environment included an Intel Core i7-10875H CPU, NVIDIA GeForce RTX 2070 GPU, 32 GB RAM, and Windows 10 64-bit operating

system. The learned models comprised four models: reference learning model (RL), standard learning model (SL), transfer learning model (TL), and material-concatenated and transfer learning model (mc-TL).

- RL is a machine learning model trained and tested on the source material. This model refers to the effective extraction of image features from the source domain.
- SL is a machine learning model trained and tested on the target material. This model can be used as a reference to compare the performances of machine learning and TL.
- TL is a transfer learning model, where a machine learning model is trained on the source material and transferred to the target material without material property concatenation. This model represents typical TL.
- mc-TL is a transfer learning model, where a machine learning model is trained on the source material and transferred to the target material with material property concatenation. This model signifies the proposed method.

This study derived the four models (RL, SL, TL, and mc-TL) separately in three cases: A. LCS (source material) to STS (target material); B. LCS to INC; C. STS to INC. In each case, the RL and SL were derived from the source and target materials, respectively. In addition, the TL and mc-TL were derived to transfer models from the source material to the target material. For example, in case A, the RL was derived from LCS, and the SL was derived from STS, whereas the TL and mc-TL were transferred from LCS to STS. The accuracy was measured using Eq. (3).

$$\text{Accuracy (\%)} = 100 \times \frac{\text{True Positive} + \text{True Negative}}{\text{the number of total data samples}} \quad (3)$$

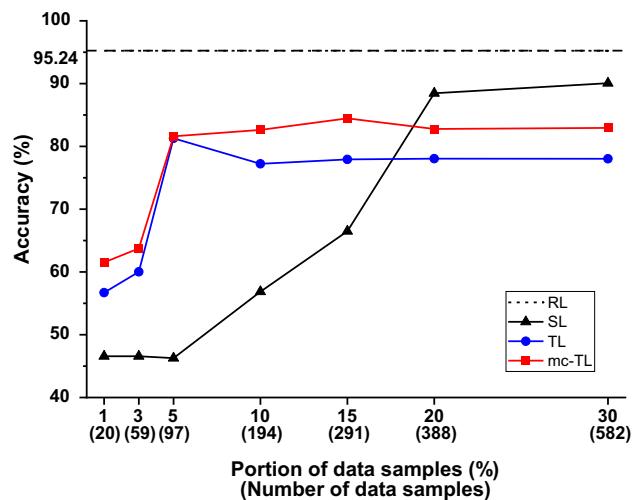


Fig. 20 Accuracy in LCS to STS

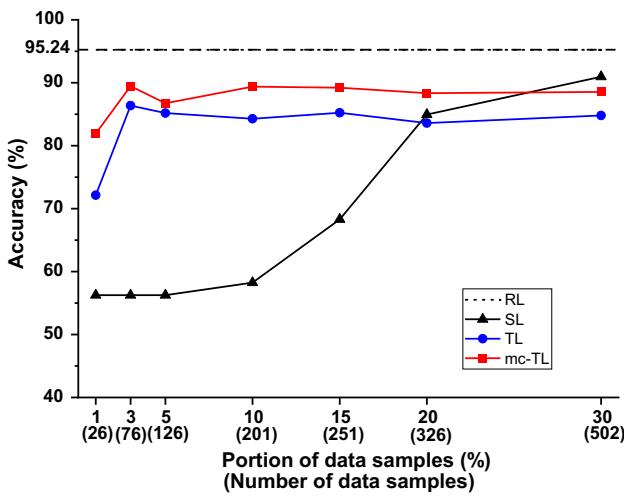


Fig. 21 Accuracy in LCS to INC

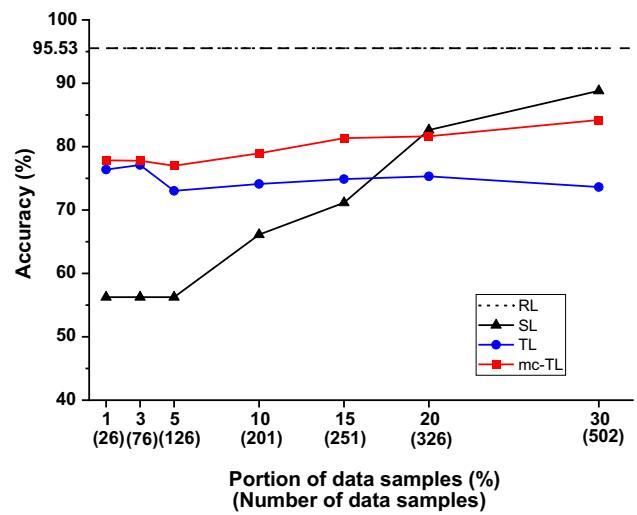


Fig. 22 Accuracy in STS to INC

A. LCS to STS: Fig. 20 presents the accuracy with respect to the increase in the number of training data samples. RL shows an accuracy of 95.24%, signifying that feature extraction is correctly performed in the source material and image features can be transferred to the target material. In SL, the accuracy scores under 66.48% for up to 15% of the whole set of training data samples but over 88.46% starting from 20% of the samples. Both accuracies increase in TL and mc-TL as the number of data samples increases. TL shows under 60.02% up to 5% of the samples and then sustains 77.22–81.28% when  $\geq 5\%$  of the samples were used. In mc-TL, the accuracy scores 61.53% and 63.74% when 1% and 3% of the samples were used, respectively. mc-TL achieved accuracies of 81.60–84.47% when  $\geq 5\%$  of the samples were used. Moreover, TL and mc-TL achieved good performance with 5% of the samples. In particular, mc-TL is more accurate than TL for all the cases. The two models outperform SL for small portions of data samples by up to 15%; however, SL has demonstrated higher accuracy for 20% or more of the samples used.

B. LCS to INC: Fig. 21 shows the accuracy results. The results show a similar pattern to those of Case A. In SL, the accuracy remains under 68.29% until 15% of the samples are used, but exceeds 84.95% when  $\geq 20\%$  of the samples are used. In TL, the accuracy is 72.14% for 1% of the samples and increases to 86.37% for 3% of the samples. However, it remains at 83.60–85.24% starting from 5% of the samples, regardless of further increases in the number of data samples. In mc-TL, the accuracy is 81.92% for 1% of the samples, but mc-TL exhibits stable accuracies of 86.74–89.47% starting from 3% of the samples. Similar to Case A, mc-TL achieves a higher accuracy than TL in all the cases.

C. STS to INC: Fig. 22 shows the accuracy results. RL achieves an accuracy of 95.53%, and the image features are applicable for transfer from the source to the target domain. In SL, the accuracy tends to increase, similar to Cases A and B; however, it does not exceed 90%, considering the maximum number of data samples. In TL, the accuracy starts increasing from 76.38% when 1% of the samples are used; however, it remains at 73.03–77.11% with all of the data samples. mc-TL achieves 76.99–78.94% when up to 10% of the samples are used. mc-TL exceeds the accuracy of 81.33% when 15% of the samples are used, whereas TL does not achieve over 80%. The accuracy of mc-TL outperforms that of TL and exhibits a slightly increasing pattern with regard to an increase in the number of data samples.

## 5.2 Discussion

### 5.2.1 Classification Evaluation

The proposed method was further compared validated using additional metrics to investigate classification performances in different viewpoints. Precision, recall, and F1-score were used as the additional metrics. Equations (4), (5), and (6) express the formula, respectively. Precision is a metric to represent how many positive predictions are correctly made over the total positive predictions, and recall is a metric to indicate how many positive predictions are correctly made over the total positive actual samples. F1-score is a metric calculated by the harmonic mean of precision and recall to weight the two metrics in a balanced way because precision and recall are in a trade-off relationship.

**Table 6** Classification evaluation results

Case	Model	Material		Portion over whole samples (number of samples)	Precision (%)	Recall (%)	F1-score	Accuracy (%)
		Source	Target					
A. LCS to STS	RL	LCS	LCS	100% (2609)	70.10	66.67	0.68	95.24
		SL	STS	1% (20)	48.71	100.00	0.65	46.57
				3% (59)	48.71	100.00	0.65	46.57
				5% (97)	51.89	100.00	0.68	46.27
				10% (194)	52.73	74.39	0.61	56.86
				15% (291)	49.29	100.00	0.65	66.48
				20% (388)	58.71	100.00	0.75	88.46
				30% (582)	61.66	100.00	0.78	90.06
		TL	LCS	1% (20)	41.07	58.00	0.48	56.71
				3% (59)	51.10	62.35	0.56	60.02
				5% (97)	71.78	83.64	0.77	81.28
				10% (194)	58.98	68.32	0.53	77.22
				15% (291)	67.40	87.02	0.75	77.93
				20% (388)	68.12	87.30	0.76	78.04
				30% (582)	68.89	85.94	0.76	78.03
		mc-TL	LCS	1% (20)	51.03	62.05	0.49	61.53
				3% (59)	58.77	46.66	0.42	63.74
				5% (97)	77.33	83.93	0.75	81.60
				10% (194)	71.19	86.25	0.77	82.62
				15% (291)	73.54	83.67	0.78	84.47
				20% (388)	70.76	84.41	0.66	82.77
				30% (582)	71.06	76.89	0.73	82.95
B. LCS to INC	RL	LCS	LCS	100% (2609)	70.10	66.67	0.68	95.24
		SL	INC	1% (26)	56.25	100.00	0.62	56.25
				3% (76)	44.28	100.00	0.62	56.25
				5% (126)	44.28	100.00	0.62	56.25
				10% (201)	56.25	100.00	0.72	58.25
				15% (251)	61.17	68.05	0.61	68.29
				20% (326)	86.14	97.20	0.91	84.95
				30% (502)	88.79	97.87	0.93	90.96
		TL	LCS	1% (26)	78.55	83.37	0.85	72.14
				3% (76)	79.04	85.82	0.82	86.37
				5% (126)	82.39	92.82	0.87	85.18
				10% (201)	81.92	99.27	0.90	84.28
				15% (251)	80.22	100.00	0.89	85.24
				20% (326)	81.70	100.00	0.90	83.60
				30% (502)	82.73	100.00	0.91	84.80
		mc-TL	LCS	1% (26)	74.59	86.12	0.80	81.92
				3% (76)	82.45	99.82	0.90	89.47
				5% (126)	77.19	95.80	0.86	86.74
				10% (201)	76.87	99.94	0.87	89.38
				15% (251)	77.56	99.94	0.87	89.22
				20% (326)	78.32	99.82	0.88	88.34
				30% (502)	78.43	99.82	0.88	88.55
C. STS to INC	RL	STS	STS	100% (1937)	95.65	93.08	0.96	95.53
		SL	INC	1% (26)	56.25	100.00	0.62	56.25
				3% (76)	44.28	100.00	0.62	56.25
				5% (126)	44.28	100.00	0.62	56.25

**Table 6** (continued)

Case	Model	Material	Portion over whole samples (number of samples)		Precision (%)	Recall (%)	F1-score	Accuracy (%)
			Source	Target				
TL	STS	INC	10% (201)		56.25	100.00	0.72	66.12
			15% (251)		63.25	68.05	0.62	71.16
			20% (326)		76.34	83.09	0.91	82.62
			30% (502)		79.87	91.59	0.83	88.81
			1% (26)		90.16	63.60	0.75	76.38
			3% (76)		67.97	89.90	0.77	77.11
			5% (126)		63.85	92.76	0.76	73.03
			10% (201)		65.43	93.49	0.79	74.11
			15% (251)		70.86	94.88	0.77	74.89
			20% (326)		74.48	96.65	0.74	75.32
mc-TL	STS	INC	30% (502)		71.41	99.27	0.73	73.63
			1% (26)		84.79	73.10	0.73	77.84
			3% (76)		69.31	87.16	0.77	77.77
			5% (126)		67.42	82.45	0.80	76.99
			10% (201)		76.79	79.18	0.83	78.94
			15% (251)		67.32	95.92	0.79	81.33
			20% (326)		68.18	95.86	0.80	81.64
			30% (502)		68.20	95.80	0.80	84.22

$$Precision (\%) = 100 \times \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (4)$$

$$Recall (\%) = 100 \times \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (5)$$

$$F1\text{-score} = 2 \times \frac{precision \times recall}{precision + recall} \quad (6)$$

Table 6 lists the classification evaluation results in the three cases (A. LCS to STS, B. LCS to INC, and C. STS to INC). Both TL and mc-TL produce stable and desirable precision values when  $\leq 15\%$  of the samples are used, compared with those of SL. However, they do not make precision improvement after the data portion becomes over 20%. In mc-TL, recall values are higher than accuracy values in general. This result implies that mc-TL can detect and identify true positive data samples nicely by reducing the probability of misclassifying positive samples as negative ones. In mc-TL, F1-score records more than 0.7 values except 1%, 3%, and 20% of the samples in case A. Note that F1-score of 0.7 is generally regarded as acceptable although this is not the absolute threshold.

### 5.2.2 Model Transferability

As shown in Figs. 20, 21 and 22, SL model exhibits an s-curve pattern as the number of data samples increases.

This model appears to be superior compared with the TL and mc-TL models and reaches up to around 90% accuracy when the portion of data samples exceeds 20%. This pattern implies that machine learning properly trains data and becomes more accurate as the amount of data increases. However, it reversely means that machine learning is viable only when sufficient data are available, particularly when low-cost materials are used.

On one hand, TL model rapidly achieves an accuracy of  $> 73\%$  when 5% of the samples are used. Then, the accuracy remains at 75–86%. This trend indicates that TL exhibits good accuracy and an increase in data samples affects the model performance less significantly. This phenomenon is common in machine learning. TL is a good substitute where data are absent or scarce; however, it shows a limitation in gaining excellence in accuracy owing to the heterogeneity of the intrinsic features between the source and target domains.

On the other hand, mc-TL model achieves accuracies of 76–89% when 5% of the samples are used. The accuracy of the model outperforms TL in each case, implying that deriving anomaly detection models is feasible even while using a small amount of training data obtained from depositing high-cost materials. In contrast to the typical model, the proposed method successfully reflects the melting and hardening characteristics of the target material by applying thermal properties to modeling.

The difference in the accuracies between TL and mc-TL is mainly induced based on the difference in transferability between them. Transferability is referred as the

generalization ability of features [34]. TL properly extracts image features to generalize the homogeneity of melting and hardening, thus producing a *common feature space*; however, there is no method to specify the thermal characteristics of materials in models. In contrast, mc-TL endows learned features with higher transferability compared to TL. This method imposes thermal properties as manufacturing knowledge features to create a *common and material-adaptive feature space*, which can implement the availability of TL and specify the discriminative thermal behaviors of materials.

### 5.2.3 Industrial Applicability

In view of the above, our mc-TL models produce acceptable classification performance in the applicability perspective because accuracy values are stably high across all the data portions and the deviations of accuracy are not massive between mc-TL and SL models (maximum deviations of 5.59%, 1.49%, and 4.59% in the three cases, respectively). These deviations can be reasonable and acceptable particularly for detecting defects in refractory materials, when data are absent or scarce because their AM processes have not been run much; otherwise, machine learning cannot be easily applied due to the cost intensiveness in data collection.

In addition, our prior studies reveal that anomaly detection models using 2D melt pool image data outperform those using 1D numerical voltage data [33], [59]. However, it is not easy to install a welding image data acquisition system in all WAAM systems owing to the cost intensiveness issue. In this situation, mc-TL can provide reasonable classification performance, taking into account cost-effectiveness with use of cheaper numerical voltage data. Thus, applicability of mc-TL can be found as an alternative anomaly detector in an environment where machine learning is unavailable owing to data absence or scarcity. Nevertheless, it is also true that controversy may arise in applying mc-TL to industries owing to the inferior accuracy compared to machine learning. This is a common phenomenon in a data sufficient environment due to the nature of TL.

## 6 Conclusion

This study proposed a material-adaptive anomaly detection method for WAAM. The following conclusions can be made from the study:

- A TL-based method was proposed to create anomaly detection models for classifying balling defects as abnormal by transferring models derived from source materials to target materials. Specifically, the proposed method differs from typical TL because it converts numerical

voltage data into voltage image data as an input to CNNs and concatenates thermal properties with image features in CNN-based modeling. The proposed method performs fine-tuning to adjust the image and material-property features of the source domain toward those of the target domain.

- Experiments were performed using a GTAW-based WAAM system. LCS, STS, and INC were used as materials, and Densenet169 was used as an image feature extractor. The proposed method generated mc-TL models that achieved accuracies of 82.95%, 89.47%, and 84.22% while transferring LCS to STS, LCS to INC, and STS to INC, respectively, outperforming typical TL models, which achieved accuracies of 78.03%, 86.37%, and 73.63%, respectively. Moreover, the method can help achieve desirable accuracy using even 3% of the data samples for the target material, rarely achieved in machine learning.

The proposed method can contribute to developing anomaly detection models for in-situ quality monitoring using TL, which has been negligibly applied for WAAM. Furthermore, material properties can be concatenated with typical image features to improve accuracy, particularly when data are scarce. Obtaining data from high-cost material fabrication is expensive and time-consuming.

The limitations of this study are as follows: First, TL is a good substitute for machine learning when data are absent or scarce, with this phenomenon demonstrated in our experiments. However, TL demonstrated the inferior accuracy compared to machine learning as training data increased. Second, the current study only uses voltage image data as a data source. This single data source may adversely affect accuracy when data are of low quality owing to sources of noise. This low data quality in a single source may constrain the achievement of a desirable accuracy because data cannot perfectly reflect the melting and solidification phenomena. Ensemble learning can help resolve this data quality issue. Ensemble learning combines more than two base learners allocated to separate data sources to improve accuracy. Each base learner trains a dataset provided from each data source, and an ensemble learner then trains prediction result data obtained from the base learners. For this study, two base learners can be derived from training the camera image data and the voltage image data. Then, ensemble learners are derived as the last anomaly detection model by training the dataset, which concatenates the probabilities of abnormality occurrences generated from the two base learners. Ensemble learning can prevent miss-classification by resolving a local optimal problem, in which a base learner can be trapped.

We plan to apply a multisource approach with multiple source domains to improve accuracy. In addition, we plan to develop a hybrid method to simultaneously use machine

learning and TL regardless of data richness or absence, thereby predicting and optimizing the quality performance in the process planning phase.

**Funding** This research was supported by the MSIT (Ministry of Science, ICT), Korea, under the High-Potential Individuals Global Training Program (No. 2021-0-01566) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). This material is based upon work supported by the National Science Foundation under Grant No. 2015693. The authors of this paper appreciate the continuous support provided by the Center for Manufacturing Research (CMR) and the Department of Manufacturing and Engineering Technology at Tennessee Technological University.

## Declarations

**Conflict of interest** The authors declare that they do not have any conflict of interest.

## References

- ISO/ASTM. (2015). ISO/ASTM 52900: Additive manufacturing—General principles—terminology.
- Gao, M., Li, L., Wang, Q., Ma, Z., Li, X., & Liu, Z. (2022). Integration of additive manufacturing in casting: Advances, challenges, and prospects. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 9, 305–322.
- Ashok, D., Mertens, A. J., & Bahubalendruni, M. V. A. R. (2023). Characterization of penetrate and interpenetrate tessellated cellular lattice structures for energy absorption. *Proceedings of the Institution of Mechanical Engineers, Part L: Journal of Materials: Design and Applications*, 237(4), 906–913.
- Rane, K., & Strano, M. (2019). A comprehensive review of extrusion-based additive manufacturing processes for rapid production of metallic and ceramic parts. *Advanced Manufacturing*, 7, 155–173.
- Lee, C. M., Woo, W. S., Baek, J. T., & Kim, E. J. (2016). Laser and arc manufacturing processes: A review. *International Journal of Precision Engineering and Manufacturing*, 17(7), 973–985.
- Kumar, A., Maji, K., & Srivastava, A. (2023). Investigations on deposition geometry and mechanical properties of wire arc additive manufactured Inconel 625. *International Journal of Precision Engineering and Manufacturing*, 24, 1483–1500.
- Pragana, J. P. M., Bragança, I. M. F., Silva, C. M. A., & Martins, P. A. F. (2023). Hybrid wire-arc additive manufacturing of conformal cooling channels: A feasibility study. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 10, 45–57.
- Wu, B., Pan, Z., Ding, D., Cuiuri, D., Li, H., Xu, J., & Norrish, J. (2018). A review of the wire arc additive manufacturing of metals: Properties, defects and quality improvement. *Journal of Manufacturing Processes*, 35, 127–139.
- Liu, J., Xu, Y., Ge, Y., Hou, Z., & Chen, S. (2020). Wire and arc additive manufacturing of metal components: A review of recent research developments. *International Journal of Advanced Manufacturing Technology*, 111, 149–198.
- Panda, B. N., Bahubalendruni, M. V. A. R., Biswal, B. B., & Leite, M. (2017). A CAD-based approach for measuring volumetric error in layered manufacturing. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 231(13), 2398–2406.
- Cunningham, C. R., Flynn, J. M., Shokrani, A., Dhokia, V., & Newman, S. T. (2018). Invited review article: Strategies and processes for high quality wire arc additive manufacturing. *Additive Manufacturing*, 22, 672–686.
- Jin, B., Tan, Y., Nettekoven, A., Chen, Y., Topcu, U., Yue, Y., & Sangiovanni-Vincentelli, A. (2019). An encoder-decoder based approach for anomaly detection with application in additive manufacturing. In *18th IEEE international conference on machine learning and applications*, Boca Raton, USA, December 16–19 (pp. 1008–1015).
- Scime, L., Siddle, D., Baird, S., & Paquit, V. (2020). Layer-wise anomaly detection and classification for powder bed additive manufacturing processes: A machine-agnostic algorithm for real-time pixel-wise semantic segmentation. *Additive Manufacturing*, 36, 101453.
- Antony, J. (2014). *Design of experiments for engineers and scientists* (2nd ed.). Elsevier.
- Kim, S. W., Kong, J. H., Lee, S. W., & Lee, S. (2022). Recent advances of artificial intelligence in manufacturing industrial sectors: A review. *International Journal of Precision Engineering and Manufacturing*, 23, 111–129.
- Meng, L., McWilliams, B., Jarosinski, W., Park, H. Y., Jung, Y. G., Lee, J., & Zhang, J. (2020). Machine learning in additive manufacturing: A review. *JOM Journal of the Minerals Metals and Materials Society*, 72, 2363–2377.
- Scime, L., & Beuth, J. (2018). A multi-scale convolutional neural network for autonomous anomaly detection and classification in a laser powder bed fusion additive manufacturing process. *Additive Manufacturing*, 24, 273–286.
- Alabi, M. O., Nixon, K., & Bote, I. (2018). A survey on recent applications of machine learning with big data in additive manufacturing industry. *American Journal of Engineering and Applied Sciences*, 11, 1114–1124.
- Lee, C., Seo, G., Kim, D. B., Kim, M., & Shin, J. H. (2021). Development of defect detection AI model for wire + arc additive manufacturing using high dynamic range images. *Applied Sciences*, 11, 7541.
- Bessa, M. A., Bostanabad, R., Liu, Z., Hu, A., Apley, D. W., Brinson, C., Chen, W., & Liu, W. K. (2017). A framework for data-driven analysis of materials under uncertainty: Countering the curse of dimensionality. *Computer Methods in Applied Mechanics and Engineering*, 320, 633–667.
- Wuest, T., Weimer, D., Irgens, C., & Thoben, K. D. (2016). Machine learning in manufacturing: Advantages, challenges, and applications. *Production and Manufacturing Research*, 4, 23–45.
- Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22, 1345–1359.
- Qiu, Y. D., Wu, J. M., Chen, A. N., Chen, P., Yang, Y., Liu, R. Z., Chen, G., Chen, S., Shi, Y. S., & Li, C. H. (2020). Balling phenomenon and cracks in alumina ceramics prepared by direct selective laser melting assisted with pressure treatment. *Ceramics International*, 46, 13854–13861.
- DebRoy, T., Wei, H. L., Zuback, J. S., Mukherjee, T., Elmer, J. W., Milewski, J. O., Beese, A. M., Wilson-Heid, A., De, A., & Zhang, W. (2018). Additive manufacturing of metallic components—Process, structure and properties. *Progress in Materials Science*, 92, 112–224.
- Khan, A., Sohail, A., Zahoor, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 53, 5455–5516.
- Zhang, Y., Hong, G. S., Ye, D., Zhu, K., & Fuh, J. Y. H. (2018). Extraction and evaluation of melt pool, plume and spatter information for powder-bed fusion AM process monitoring. *Materials and Design*, 156, 458–469.

27. Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*. <https://doi.org/10.1186/s40537-019-0197-0>

28. Panda, B. N., Bahubalendruni, M. V. A. R., & Biswal, B. B. (2015). A general regression neural network approach for the evaluation of compressive strength of FDM prototypes. *Neural Computing & Applications*, 26, 1129–1136.

29. Mojahed Yazdi, R., Imani, F., & Yang, H. (2020). A hybrid deep learning model of process-build interactions in additive manufacturing. *Journal of Manufacturing Systems*, 57, 460–468.

30. Lyu, J., & Manoochehri, S. (2021). Online convolutional neural network-based anomaly detection and quality control for fused filament fabrication process. *Virtual and Physical Prototyping*, 16(2), 160–177. <https://doi.org/10.1080/17452759.2021.1905858>

31. Yan, H., Grasso, M., Paynabar, K., & Colosimo, B. M. (2022). Real-time detection of clustered events in video-imaging data with applications to additive manufacturing. *IIE Transactions*, 54, 1–28.

32. Segura, L. J., Wang, T., Zhou, C., & Sun, H. (2021). Online droplet anomaly detection from streaming videos in inkjet printing. *Additive Manufacturing*, 38, 101835.

33. Cho, H. W., Shin, S. J., Seo, G. J., Kim, D. B., & Lee, D. H. (2022). Real-time anomaly detection using convolutional neural network in wire arc additive manufacturing: Molybdenum material. *Journal of Materials Processing Technology*, 302, 117495.

34. Yang, Q., Zhang, Y., Dai, W., & Pan, S. J. (2020). *Transfer learning*. Cambridge University Press.

35. Bang, S. H., Ak, R., Narayanan, A., Lee, Y. T., & Cho, H. (2019). A survey on knowledge transfer for manufacturing data analytics. *Computers in Industry*, 104, 116–130.

36. Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision pattern recognition*, Columbus, OH, USA, June 23–28 (pp. 1717–1724).

37. Shao, S., McAleer, S., Yan, R., & Baldi, P. (2018). Highly accurate machine fault diagnosis using deep transfer learning. *IEEE Transactions on Industrial Informatics*, 15, 2446–2455.

38. Guo, L., Lei, Y., Xing, S., Yan, T., & Li, N. (2018). Deep convolutional transfer learning network: A new method for intelligent fault diagnosis of machines with unlabeled data. *IEEE Transactions on Industrial Electronics*, 66, 7316–7325.

39. Sun, C., Ma, M., Zhao, Z., Tian, S., Yan, R., & Chen, X. (2018). Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing. *IEEE Transactions on Industrial Informatics*, 15, 2416–2425.

40. Ferguson, M. K., Ak, R., Lee, Y. T. T., & Law, K. H. (2018). Detection and segmentation of manufacturing defects with convolutional neural networks and transfer learning. *Smart and Sustainable Manufacturing Systems*. <https://doi.org/10.1520/SSMS20180033>

41. Imoto, K., Nakai, T., Ike, T., Haruki, K., & Sato, Y. (2018). A CNN-based transfer learning method for defect classification in semiconductor manufacturing. In *International symposium on semiconductor manufacturing (ISSM)*, Tokyo, Japan, December 10–11.

42. Pan, Y., Mei, F., Miao, H., Zheng, J., Zhu, K., & Sha, H. (2019). An approach for HVCB mechanical fault diagnosis based on a deep belief network and a transfer learning strategy. *Electrical Engineering Technology*, 14, 407–419.

43. Zellinger, W., Grubinger, T., Zwick, M., Lughofner, E., Schöner, H., Natschläger, T., & Saminger-Platz, S. (2020). Multi-source transfer learning of time series in cyclical manufacturing. *Journal of Intelligent Manufacturing*, 31, 777–787.

44. Wang, P., & Gao, R. X. (2020). Transfer learning for enhanced machine fault diagnosis in manufacturing. *CIRP Annals*, 69, 413–416.

45. Gong, Y., Shao, H., Luo, J., & Li, Z. (2020). A deep transfer learning model for inclusion defect detection of aeronautics composite materials. *Composite Structures*, 252, 112681.

46. Michau, G., & Fink, O. (2021). Unsupervised transfer learning for anomaly detection: Application to complementary operating condition transfer. *Knowledge Based System*, 216, 106816.

47. Liu, B., Zhang, Y., Lv, J., Majeed, A., Chen, C. H., & Zhang, D. (2021). A cost-effective manufacturing process recognition approach based on deep transfer learning for CPS enabled shop-floor. *Robotics and Computer-Integrated Manufacturing*, 70, 102128.

48. Kim, Y. M., Shin, S. J., & Cho, H. W. (2022). Predictive modeling for machining power based on multi-source transfer learning in metal cutting. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 9, 107–125.

49. Marei, M., El Zaatar, S. E., & Li, W. (2021). Transfer learning enabled convolutional neural networks for estimating health state of cutting tools. *Robotics and Computer-Integrated Manufacturing*, 71, 102145.

50. Liu, J., Guo, F., Gao, H., Li, M., Zhang, Y., & Zhou, H. (2021). Defect detection of injection molding products on small datasets using transfer learning. *Journal of Manufacturing Processes*, 70, 400–413.

51. Tang, Y., Dehaghani, M. R., & Wang, G. G. (2023). Review of transfer learning in modeling additive manufacturing processes. *Additive Manufacturing*, 61, 103357.

52. Ho, S., Zhang, W., Young, W., Buchholz, M., Al Jufout, S. A., Dajani, K., Bian, L., & Mozumdar, M. (2021). DLAM: Deep learning based real-time porosity prediction for additive manufacturing using thermal images of the melt pool. *IEEE Access.*, 9, 115100–115114. <https://doi.org/10.1109/ACCESS.2021.3105362>

53. Zhu, X., Jiang, F., Guo, C., Xu, D., Wang, Z., & Jiang, G. (2023). Surface morphology inspection for directed energy deposition using small dataset with transfer learning. *Journal of Manufacturing Processes*, 93, 101–115.

54. Nannapaneni, S., Mahadevan, S., & Rachuri, S. (2016). Performance evaluation of a manufacturing process under uncertainty using Bayesian networks. *Journal of Cleaner Production*, 113, 947–959.

55. Somasundaram, A., & Reddy, U. S. (2016). Data imbalance: Effects and solutions for classification of large and highly imbalanced data. In *Proceedings of the 1st international conference on research in engineering, computers and technology (ICRECT)*, Tiruchirappalli, India, September 8–10 (pp. 28–34).

56. Xing, Z., Pei, J., & Keogh, E. (2010). A brief survey on sequence classification. *ACM SIGKDD Explorations Newsletter*, 12, 40–48.

57. Zhao, B., Lu, H., Chen, S., Liu, J., & Wu, D. (2017). Convolutional neural networks for time series classification. *Journal of Systems Engineering and Electronics*, 28, 162–169.

58. Koprinska, I., Wu, D., & Wang, Z. (2018). Convolutional neural networks for energy time series forecasting. 2018, Rio de Janeiro, Brazil, July 8–13 (pp. 1–8).

59. Kim, E. S., Lee, D. H., Seo, G. J., Kim, D. B., & Shin, S. J. (2023). Development of a CNN-based real-time monitoring algorithm for additively manufactured molybdenum. *Sensors & Actuators: A. Physical*, 352, 114205.

60. Wang, C., Tan, X. P., Tor, S. B., & Lim, C. S. (2020). Machine learning in additive manufacturing: State-of-the-art and perspectives. *Additive Manufacturing*, 36, 101538.

61. Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *IEEE conference on computer vision and pattern recognition (CVPR)*, Honolulu, USA, 2017, July 21–26.

62. Zheng, L., Zhao, Y., Wang, S., Wang, J., & Tian, Q. (2016). Good practice in CNN feature transfer. *CVPR*. [arXiv:1604.00133](https://arxiv.org/abs/1604.00133).
63. MatWeb. Retrieved March 27, 2021, from <http://www.matweb.com/index.aspx>.
64. Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B., & Liang, J. (2016). Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Transactions on Medical Imaging*, 35, 1299–1312.
65. Hadhrami, E. A., Mufti, M. A., Taha, B., & Werghi, N. (2018). Transfer learning with convolutional neural networks for moving target classification with micro-Doppler radar spectrograms. In *International conference on artificial intelligence and big data (ICAIBD)*, Chengdu, China, 2018, May 26–28.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.



**Seung-Jun Shin** is an Associate Professor in School of Interdisciplinary Industrial Studies at Hanyang University. He received his bachelor degree (2002) in Mechanical Engineering at Korea University and his master degree (2005) and Ph.D. degree (2010) in Department of Industrial and Management Engineering at POSTECH. He worked as a senior engineer in Samsung Electronics, a senior consultant in Samsung SDS, a guest researcher at National Institute of Standards and Technology, and an assistant professor in Pukyong National University. His research interests include cyber-physical production systems, manufacturing data analytics, industrial standards, and environmentally-conscious manufacturing.



**Ju-hong Lee** received his bachelor degree (2019) in Department of Systems Management and Engineering at Pukyong National University and his master degree (2022) in Department of Industrial Data Engineering at Hanyang University. He is working as a data engineer in LG Display. His research interests include cyber-physical production systems, manufacturing data analytics, smart factory, and digital twin.



**Sainand Jadhav** is a Ph.D. candidate in the Department of Mechanical Engineering at Tennessee Technological University, USA. He obtained his M. Tech. degree in Mechanical Engineering (CAD/CAM) from Bharati Vidyapeeth Deemed University, India. His research interest is in metal additive manufacturing, material characterization, robotic welding and WAAM process development for bimetallic structures.



**Dr. Duck Bong Kim** received his MS and PhD degrees at Gwangju Institute of Science and Technology (GIST), Republic of Korea, in 2006 and 2011. He worked as a guest researcher at National Institute of Standards and Technology (NIST) from 2011 to 2016. He has been a professor in the Department of Manufacturing and Engineering Technology at Tennessee Technological University since 2016. His research interests include metal additive manufacturing, smart manufacturing, and data analytics.