Digital Twin of Intelligent Small Surface Defect Detection with Cyber-Manufacturing Systems

YIRUI WU, College of Computer and Information, Hohai University, China
Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, China
HAO CAO, College of Computer and Information, Hohai University, China
GUOQIANG YANG, Key Laboratory for Novel Software Technology, Nanjing University, China
TONG LU, Key Laboratory for Novel Software Technology, Nanjing University, China
SHAOHUA WAN*, Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, China

With the remarkable technological development in cyber-physical systems, industry 4.0 has evolved by a significant concept named as digital twin (DT). However, it's still difficult to construct relationship between twin simulation and real scenario considering dynamic variations, especially when dealing with small surface defect detection tasks with high performance and computation resource requirement. In this paper, we aim to construct cyber-manufacturing systems to achieve a DT solution for small surface defect detection task. Focusing on DT based solution, the proposed system consists of an Edge-Cloud architecture and a surface defect detection algorithm. Considering dynamic characteristics and real-time response requirement, Edge-Cloud architecture is built to achieve smart manufacturing by efficiently collecting, processing, analyzing, and storing data produced by factory. A deep learning based algorithm is then constructed to detect surface defeats based on multi-modal data, i.e., imaging and depth data. Experiments show the proposed algorithm could achieve high accuracy and recall in small defeat detection task, thus constructing DT in cyber-manufacturing.

CCS Concepts: • **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

Additional Key Words and Phrases: defect detection; cyber manufacturing; digital twin; 3D point cloud;

1 INTRODUCTION

Considering that most manufacturing operations heavily depend on experienced persons, both small and large equipment manufacturers have an increasing demand for the deployment of intelligent manufacturing machines

*The corresponding author.

Authors' addresses: Yirui Wu, wuyirui@hhu.edu.cn, College of Computer and Information, Hohai University, No. 8, Focheng West Road, Nanjing, China, 210024
and Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Qianjin Road, Changchun, China, 210024; Hao Cao, College of Computer and Information, Hohai University, No. 8, Focheng West Road, Nanjing, China, haocaohh@gmail.com; Guoqiang Yang, Key Laboratory for Novel Software Technology, Nanjing University, No. 8, Focheng West Road, Nanjing, China, haocaohh@gmail.com; Tong Lu, Key Laboratory for Novel Software Technology, Nanjing University, No. 8, Focheng West Road, Nanjing, China, lutong@nju.edu.cn; Shaohua Wan, Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Guanlan Road, Shenzhen, Guangdong, China, shaohua.wan@uestc.edu.cn.

1533-5399/2022/11-ART $15.00
https://doi.org/10.1145/3571734

with affordable price and reliable technologies. Inspired by Cyber-Physical systems, Cyber-Manufacturing (CM) concept thus appears, which aims to link between significant elements, intertwine industrial big data and smart analytics, discovering and comprehending invisible issues for decision making. As the core technologies of CM, Internet-of-Things (IoT) and predictive analytics have advanced to obtain an emerging virtual representation solution named as digital twin (DT).

With advancement of artificial intelligence and big data analysis, DT enables to collect data from physical space through conventional devices, and make rapid analysis and real-time decisions on the collected data, which ensures the execution of automated systems. More importantly, DT couples collaboration between the physical and virtual worlds equipped with Cyber-Manufacturing systems (CMS), enabling manufacturing operations to integrate resources on a global scale and develop extensive cooperation [15, 19].

Essentially, how to facilitate DT in smart manufacturing remains an open question, calling for a systematic methodology to build a networked data-rich environment, and transform raw data into meaningful and actionable operations. In this paper, we focus on constructing a DT solution for small surface defeat detection task, which scans product surface by sensors, transmits usable information, detect categories and locations of surface defects in virtual space, and determines further operations in physical world. The reason that defeat detection task is suitable to build DT lies in two aspects. First, since surface defeat detection exists in high-risk workshops like tires and chips, the high density of personnel may cause safety hazards, setting strict requirements on the distribution of personnel in different areas. Once human workers have been recognized as essential factor, their natural undeterminate characteristics may harm manufacturing yield rate. Second, computer vision technology has been successfully applied for surface defect detection in relative simple workshops [24], proving possibilities to further improve it for high reusability, reliability and predictability by latest development of IoT and AI.

To construct DT for defeat detection task in a distributed and collaborative environment, there essentially exists two urgent needs, that is (1) hardware and software architecture that efficiently collect and analyze large volumes of data generated from scanners and (2) algorithms that effectively diagnose the identified defects, and forecast maintenance activities to minimize unexpected loss. Following such requirements, we further analyze the limitations considering its inherently problematical properties.

From perspective of hardware and software architecture, there exists a lack of affordable sensing technologies that can be readily integrated into manufacturing systems. Choosing proper sensors from few candidates to keep balance between banquet and performance thus becomes an important task, in order to lay solid foundation of data acquirement for DT. Due to the existence of interference appearances such as patterns, stains, and reflections, it's difficult to capture small deformation of surface only with image data. We propose to capture abundant surface information to achieve reliable detection results by both image and depth cameras, thus forming 2D and 3D big data for further analyzing. DT with CPS requires to analyze multi-physics data streams with high speed, high volume, and high variety, in real-time, thus demanding information and communication technology (ICT) infrastructures and parallel algorithms to equip with sufficient computational capacity and bandwidth. Moreover, high-precision 3D point cloud data brings pressure on computing resources for deformation detection. Since it's too expensive to build high performance computing clusters for training of deep learning models, how to proper involve edge and cloud infrastructures to enable remote sensing and load balance for real-time detection remains a challenge. With the idea to build an expandable paradigm for DT with CPS, we design a simple but effect Edge-Cloud architecture to efficiently collect, process, analyze, and store big manufacturing data.

From perspective of detection algorithm, a robust learning algorithm is necessary to tell what and where is the defect accurately and quickly, due to the different defeat classes, surface background, and illuminations. However, single-mode data, i.e., either 2D or 3D data, would lead to non-robust performance, proved by large deviation between training and testing performance. Therefore, an efficient multi-modal feature fusion mechanism should be addressed to seamlessly integrate the collected multi-dimensional sensing data for dynamic evaluations. To achieve predictable and reliable DT with complexity of dynamic environment, we design a deep learning scheme to effectively distinguish between defeats and non-defeats. Moreover, weak deformation defects might be similar with patterns of texture or background, which leads to non-overlapping false detection results. Considering high cost for failure cases in detection, it's a wise option to conduct post-processing from another but effective analyzing view, i.e., morphological operator based on patterns extracted from pre-collected samples, which greatly improves precision performance by suppressing non-overlapping detection results.

The main contributions of this paper are as follows:

- A framework of intelligent small surface defect detection for DT is designed with CMS technologies, which monitors product conditions and generate predictive analytic with dynamic and real-time characteristics.
- A simple but effect Edge-Cloud architecture is built, which not only connects sensors and computation devices for 2D and 3D big data collecting, but also enables remote sensing and load balance for real-time detection.
- A deep learning based small surface defect detection algorithm is proposed, in which features of multi-modal data are extracted and fused as abundant information source for reliable analyzing.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 presents an overview of the intelligent small surface defect detection framework. Details of the proposed deep learning algorithm, including detection goal for smart manufacturing, structure design of intelligent small surface defect detection algorithm. Section 5 presents the experimental results and discussions. Finally, Section 6 concludes the paper.

2 RELATED WORK

In this section, several related issues, including Digital Twin in Cyber-Manufacturing, and surface defect detection algorithm, are reviewed, respectively.

2.1 Digital Twin in Cyber-Manufacturing

Enterprises of different sizes in various countries undertake the same manufacturing activities, forming a complex and decentralized manufacturing network. Built on the basis of network, CM refers to the use of high performance computing, optimization, simulation, sensing technology, and data analytics to create innovative products [31]. As one of the most promising technologies for smart manufacturing, DT reflects the evolution of the whole life cycle of physical entities by integrating multi-disciplinary, multi-physical quantity, multi-scale and multi-probability simulation processes, and realizes the synchronous mapping of dynamic physical world in digital space. Inspired by the robotic digital twin, value-driven and other similar methods can solve the problem of data sensing in dual environments by minimizing the changes between the physical and the virtual spaces, thus achieving effective simultaneous mapping of physical and digital space [17, 30]. Essentially, the introduction of DT has greatly promoted the development of cyber-manufacturing. For example, based on DT-based virtual simulations in CPS, complex and varying environment factors can be effectively analyzed and thus regulated during manufacturing. Meanwhile, CMS interface can be used for data insertion and data visualization during DT in a data-driven way [38]. Moreover, CMS technologies can be adopted to promote the realization of DT space by collecting large volumes of real-time data or building large-scale predictive models for significant advances.

As a successful example for DT with CPS, Zhou et al. [36, 39, 42] propose that equipment, product, and operator are three basic environmental parameters, where he builds DT for small object detection in smart manufacturing, analyzing and estimating the dynamic characteristics and real-time changes from physical manufacturing space to virtual space. Since existing monitoring systems and prognostics approaches are not capable to support the construction of DT, Wu et al. [34] propose a new computational framework for diagnosis and prognosis, which enables remote real-time sensing, monitoring, and scalable high performance computing, utilizing wireless sensor networks, cloud computing, and machine learning as core inventions from CM. Focusing on intrusion detection, Wu et al. [35] propose a conceptual system to detect cyber-physical intrusions in CMS, where physical data from the manufacturing process level and production system level are integrated with cyber data from network-based and host-based IDSs, meanwhile the correlations between the cyber and physical data are analyzed by machine learning for intrusion detection.

Besides smart manufacturing, DT has been widely used in other domains, such as smart city, medical analysis [8], and hydrology construction [4]. In 2010, NASA propose the goal of DT in space technology is to halve the maintenance cost and ten times extend service life of aircraft by 2035. The European Space Agency launch its DT Earth project with the intention of realizing dynamic and interactive natural twin systems. Meanwhile, Bauer et al. [2] publish a study in Nature for DT Earth construction by collaborative optimization between observational data and physical models. China begins its DT city construction of Xiongan New Area, in which 25.4-square-kilometer central business district has realized digital mapping of urban elements and dynamic supervision of building projects. For the multi-source data collected in smart cities, Li et al. [20] introduce a deep learning algorithm for big data analysis (BDA), and propose a distributed parallel strategy of convolutional neural network (CNN). Through digital twins (DTs) and multi-hop transmission technology, they build a DL-based smart city DTs multi-hop transmission IoTBDA system, and further simulate the performance of the system, enabling the transformation of smart cities shift to granular governance and secure data processing. CARES research team [40] develop the UK Digital Twin platform, which utilizes knowledge graph and agent technology to analyze multi-disciplinary big data and combine ontology characterized conceptual instances, the mirror world and parallel world thus being established in the virtual space. For example, Samah et al. [1] propose MMSUM Digital Twins, i.e., a summarization framework that is capable of generating a multi-view multi-modal summary for sporting events in real time to effectively summarize the development process of sports events and focus on fans' reactions and subjective opinions. Through sentiment analysis to track fans' state of mind, MMSUM can complete the evaluation of the generated multi-view summaries. Furthermore, digital twins can also be combined with other technologies to solve practical problems. To reconcile the conflict between privacy preservation and data training in air-ground networks, Sun et al. [23, 32] consider dynamic digital twin and federated learning for air-ground networks where a drone acts as the aggregator based on the networks captured by digital twin. In this model, the digital twin provides a virtual representation for air-ground network to reflect time-varying states. Moreover, considering the varying digital twin deviations and network dynamics and network dynamics, they design a dynamic incentive scheme to adaptively adjust the selection of the optimal clients and their participation level.

2.2 Surface Defect Detection Algorithm

We classify current surface defect detection algorithm into two categories based on the input mode, i.e., images or 3D point cloud.

Surface Defect Detection Algorithm Based on Images. Traditional defect detection algorithms mainly rely on manually designed features, like SIFT and ORB. However, they generally suffer from poor robustness when facing complex pattern hidden images. Inspired by remarkable distinguish capability, deep learning methods have become the mainstream for surface defect detection.

Early, Faghih-Roohi et al. [10] use ReLU for the activation function and evaluate several network sizes for the specific problem of classifying rail-surface defects. Later, Racki et al. [29] propose a more efficient network to explicitly perform the segmentation of defects, where they design an additional decision network on top of the features from the segmentation network to perform a per-image classification of a defects presence, improving classification accuracy for surface defect detection. Afterwards, Lin et al. [21] propose LEDNet for defect detection on LED chips with 30,000 low-resolution images, where their network follows general structure of AlexNet by replacing fully connected layers with incorporates class-activation maps (CAMs). Inspired by Gaussian heatmaps to characterize keypoints in pose estimation applications, CornerNet [18] is proposed that uses top-left and bottom-right corners of objects to construct Gaussian heatmaps for object representation. On the basis of CornerNet, CenterNet [41] uses the center point and size of object instead, where a modified version of CenterNet [16] successfully detects tile crack defects with high mean average precision (mAP) performance.

In manufacturing process, weak feature representation of defects would cause defects to be submerged by noise and background. To avoid this, He et al. [13] propose a system for steel plate defect detection, which uses a baseline convolutional neural network (CNN) and a multi-level feature fusion network (MFN) to combine multiple levels of features, greatly enhancing weak features to represent defect details. Despite weak features, small training dataset remains difficulty, since too few training sample prone to be overfitting for deep learning structure. To mitigate overfitting problem, Tabernik et al. [33] present a segmentation-based deep-learning architecture that is designed for the detection and segmentation of surface anomalies, where the architecture enables the model to be trained using a small number of samples, thus being practical for real-scene applications. In order to solve the time-consuming problem of deep learning models in automatic optical metal defect detection systems, Lin et al. [22] used Spearman rank correlation, Pearson correlation and Kendall correlation to replace the evaluation methods in traditional detection models, and achieve better performance.

Surface Defect Detection Algorithm Based on 3D Point Cloud. We classify current methods into three categories, i.e., multi-view based, Voxelization-based, and raw-data based. Multi-view based methods transform disordered, unstructured 3D point cloud data to structured, two-dimensional data with bird-eye and front view through projection and interpolation, thus detecting surface defeats by regarding transformed data as images [5, 25]. Later, MV3D [6] is proposed with two stages, namely 3D Proposal Network and Region-based Fusion Network. Specially, the former network first extracts features from the input bird's-eye view, front view, and RGB images, and then obtains a large number of candidate 3D bounding box predictions that may contain objects from the obtained feature maps. By integrating candidate features from different sources into the same dimension using RoI pooling, the later network fuse features to accurately predict the class and 3D bounding box of the object.

Voxelization-based defect detection algorithms aggregate unstructured points into structured voxel representations, meanwhile maintaining three-dimensional information [9]. For example, SECOND [37] designs 3D coefficient convolution operation, which effectively improves the speed of voxel-based 3D point cloud detection algorithm, meanwhile solving the problem of empty voxels in transformed data. Different from information loss caused by the previous two kinds of methods, detection methods based on raw-data designs to directly extract the structured multi-dimensional feature data from the original point cloud, such as Hierarchical features [11, 26, 27]. For example, PointNet++ [28] design a local feature extraction module, which perform feature extraction by downsampling on the original point cloud. Multiple features of different receptive fields are then obtained by cascading the set, where the last layer outputs the global features for accurate defeat detection. Compared with the method using single modal data and common feature fusion methods, our method can more effectively fuse the features extracted from the depth map and pseudo color map, and dynamically adjust the fusion weight between the feature relations of the two maps.

Moreover, some other methods, such as radar, can be applied to surface defect detection. Cheng et al. [7] propose a radar-vision fusion based method for small surface object detection, which adopts a novel representation format of millmeter wave radar point cloud. By fusing the multi-scale features of RGB images and radar data, the method effectively improves the accuracy and robustness of water surface precision measurement and achieves advanced performance.

3 THE PROPOSED FRAMEWORK

We introduce how to effectively monitor product conditions and generate predictive analytic with dynamic and real-time characteristics in this section. To fulfil these high standard requirements, we design a framework of

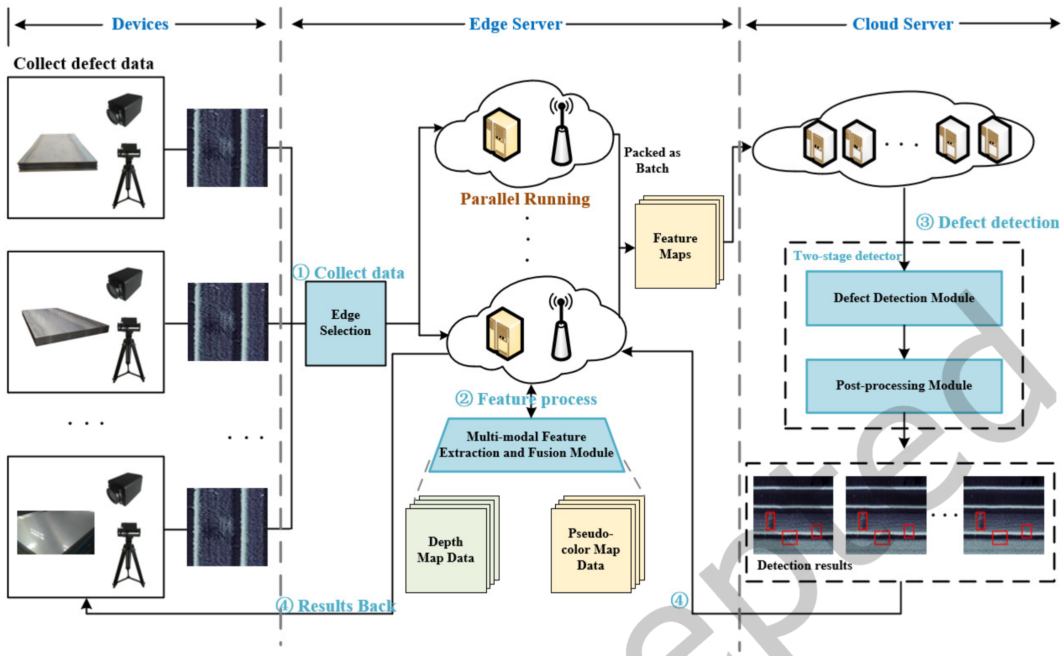


Fig. 1. Framework design of the proposed Edge-Cloud architecture for smart manufacturing, which enables interaction between physical manufacturing environment and virtual space via steps of collecting data, feature processing, defect detection and results back for improvement.

intelligent small surface defect detection for DT with CMS technologies. Specifically, we build an Edge-Cloud architecture to collect 3D point cloud data of product surface through 3D scanners, meanwhile keeping load balance in either cloud or edges for high computation capacity. Then, we offer descriptions on design of defect detection algorithm driven by the proposed edge-cloud architecture. Afterwards, an overview on structure design of the proposed intelligent small surface defect detection algorithm is proposed to perform manufacturing detection task. Finally, we will demonstrate processing steps of framework including feature extraction and fusion, detection, and post-processing, which move towards the goal of building DT with CPS technologies.

3.1 Edge-Cloud Architecture for Smart Manufacturing

Considering dynamic characteristics and real-time response requirement, we construct a simple and effective Edge-Cloud architecture for smart manufacturing, which is shown in Fig. 1. It enables remote sensing, load balance, and results sending back for improvement through the mutual mapping and timely interaction between physical manufacturing environment, i.e., factory and virtual space.

To reach such goals, the proposed architecture requires to accurately describe the proximity of the digital model to the physical model, where edge server close to the collecting devices are capable to meet these requirements. More precisely, edge server in Fig. 1 can quickly respond to the variations of physical products, thus dynamically adjusting the whole framework for better performance. The proposed architecture is thus designed with sensing devices, edge servers and cloud servers, which effectively and automatically collects, processes, analyzes and stores big data produced by stream lines of factory.

Algorithm 1 Design of Edge-driven defect detection algorithm.

Require: Collected data S
Ensure: Defect detection results P
while Collection device is working **do**
  if Obtain the data from both 2D and 3D scanners **then**
    Upload the data S to edge servers
    Extract features F_{2d} and F_{3d} from 2D and 3D input, respectively
    Fuse features F_{2d} and F_{3d} to obtain F
    Upload F to cloud server
    Obtain defect detection results P in cloud server based on F
    Return P to first edge and then sensing devices(workers) for pipeline adjustment
  else
    Wait for new collected data
return P

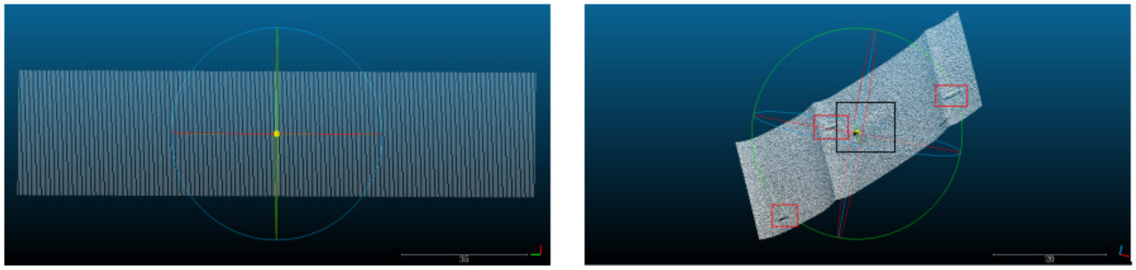
Specifically, to reduce the pressure of data transmission, we transfer the collected data captured by sensing devices to the nearest edge servers for feature processing through edge selection. Afterwards, the extracted feature are further transmitted to the cloud server for defect detection, which generally requires high computation cost via deep learning algorithms. Finally, the detection results are returned to the edge servers, guiding production activities for promotion in the factory.

Since edge-cloud architecture is a physically distributed and logically collaborative system, the proposed framework ensures capability by significantly increasing computing and storing capacity without purchasing expensive devices, meanwhile solving the limitations of local collection and processing equipments. Moreover, computing and storage pressure is effectively dispersed to several edge servers and cloud server, where the short distance between edge servers and sensing devices guarantee real-time synchronization between the physical manufacturing environment and virtual space, thus alleviating the unified management workflow of traditional automatic algorithms. Last but not least, design of feature processing on physically closer edge servers can greatly reduce the size of transferred data, reducing latency of data transmission from edge to cloud. Since cloud could undertake computationally intensive workloads due to its sufficient computing and storing resources, we employ detection and post-processing in cloud for reliable and costly analyzing.

3.2 Design of Edge-driven Defect Detection Algorithm

To fit with the proposed Edge-Cloud architecture for smart manufacturing, we modify the general steps of defect detection algorithm for better performance. As illustrated in Fig. 1 and Algorithm 1, the proposed edge-driven defect detection algorithm consists of four steps: collecting data S transmitted from sensing devices to edge servers, extracting and fusing feature map F transmitted from edge servers to cloud servers, detecting defects results P transmitted from cloud to edge, then edge to sensing devices. More precisely, if a worker working with sensing devices tries to know defect detection results of current producing product in pipeline for timely adjustment, the whole process can be described as follows:

1) Workers have options to upload their captured data probably containing small surface defect to edge server. If they choose yes to upload, sensing equipments including 2D cameras and 3D scanners will timely collect surface data on the corresponding workpiece through the gateway. Then, the data will be transmitted to the nearest edge server based on certain selection rules for edge selection.



(a) Rendering from a top-down **view** (b) Rendering **from the best view to find defects**

Fig. 2. A sample of small surface defect, where we render it from different views to better find defects.

2) The edge server employs multi-modal feature extraction and fusion module to generate feature map based on the uploaded data, which provides both 2D and 3D analysis for distinguish feature desorption. After extraction and fusion, feature map is transmitted to cloud server for further detection.

3) The cloud server employs defect detection and post-processing modules to achieve accurate detection results based on the uploaded feature maps. Both modules are designed to obtain high recall performance, thus guaranteeing non-existence of valid products during smart manufacturing.

4) The detection results are returned to first edge servers and then sensing devices, where resulting images with bounding boxes to intuitively show small surface defects can be used by workers for further determination. Once the worker standing by the sensing devices is notified with defects on current product in real time, he or she can simple abandon this product or half the whole pipeline to investigate reason for defected production.

It's worthy noting that multiple sensing devices or workers, who locate in different factories and are willing to share data, can upload data to compensate for data scarcity with the proposed edge-cloud architecture, thus greatly improving accuracy performance of defect detection model trained in the cloud. Such benefits of non-local unconstraint and iteratively optimization allows grouped factories to build DT with more confidence and patience. Besides, workers have the right to choose whether uploading data to cloud or not, thus opting out of sharing data at any time. If the data is private, workers can choose to only upload the collected data to local private edge servers for defect detection services, thus ensuring privacy of users and security of data.

3.3 Structure Design of Intelligent Small Surface Defect Detection Algorithm

In this section, we firstly analyze requirements to design intelligent small surface defect detection algorithm for DT, and then offer descriptions on overall structure design of the proposed algorithm.

Requirement Analysis. Facing challenge of recognizing defects in complex industrial scenario, it's difficult for general algorithms to keep consist performance in both easy and hard cases, due to their data-driven property. By investigating manual detection process complicated by lighting, observation and hand touch, it's of great significance to learn how to achieve robust and convinced detection results. We show 3D point cloud data of small surface defects renderings from different views in Fig. 2. It's noted that we can only observe texture of the workpiece under a top-down view in Fig. 2(a). Meanwhile, a skilled worker can quickly find the best view for defect detection as shown in Fig. 2(b), where red and black boxes mark intrinsic shape protruding and surface deformation defect, respectively.

Based on former analysis, we briefly list requirements of algorithm design to construct DT for smart manufacturing.

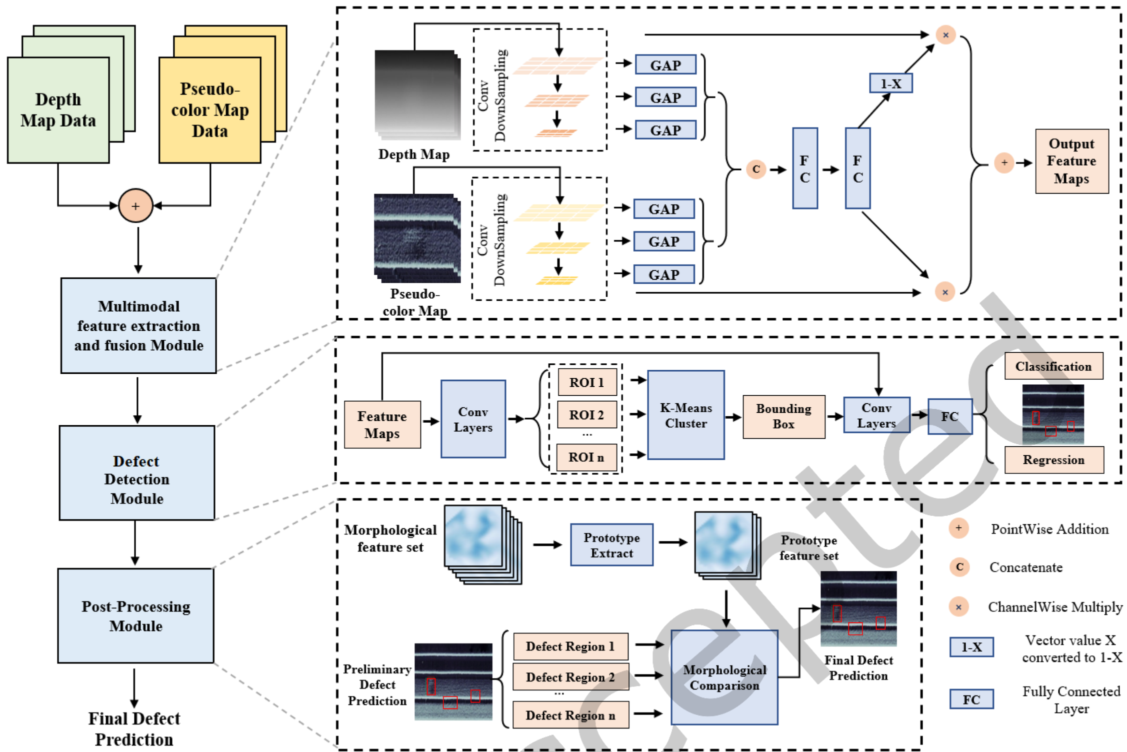


Fig. 3. Workflow of the proposed intelligent small surface defect detection algorithm.

- High recall performance. It's wise to adopt multiple modalities for defect detection with two reasons, 1) weak deformation in depth map lead to miss detection results and 2) severe deformation background in pseudo-color map is easily to be amplified in rendering process, thus resulting in wrong detection results. Moreover, Fig. 2 proves drawbacks of using single modality, which ignores much information of defects.
- Low False Rejection Rate. Most existing defect detection algorithms suffer from bias of training data, which refers to large deviation from training samples and real product samples, resulting in high false rejection rate, especially in non-overlapping detection. Since most of post-processing algorithms like NMS [12], softNMS [3], and softerNMS [14], fails in handling non-overlapping detection errors, it's a high priority requirement to design novel post-processing algorithms to eliminate such errors.

Algorithm Design. We show structure design of the proposed intelligent small surface defect detection algorithm in Fig. 3, which includes general steps of multi-modal feature extraction and fusion, defect detection and post-processing within the proposed edge-cloud architecture. It's noted that we design the whole defect detection algorithm with three stages following the classic idea of Faster-RCNN structure, i.e., feature extraction, detection and post-processing. Moreover, we modifies it to fit with multi-modal input, and improves it in post-processing with morphology operations to further improve detection performance on extremely small surface defects.

Aiming to improve low recall performance caused by only using single modality of surface defect samples, the first step of feature extraction and fusion adaptively defines fusion weights for either 2D or 3D modality based on

Algorithm 2 Design of Intelligent small surface defect detection algorithm.

Require: Input depth data S_d^i, color image data S_c^i, where i refers to the ith batch
Ensure: Detection bounding boxes B^i, their corresponding confidence scores C^i
1: Extract multiple modality feature maps F_d^i and F_c^i based on S_d^i and S_c^i respectively, represented by Eq. 2
2: Obtain F_f^i by fusing F_d^i and F_c^i with a weighting scheme, represented by Eq. 3
3: Obtain set of detection bounding boxes B^i and their corresponding confidence scores C^i with the proposed defect detection module, represented by Eq. 1
4: Suppress jth bounding box B_j^i by decreasing C_j^i, if distance in feature map between $F_{f,j}^i$ and pre-extracted prototypes F_p is larger than threshold, represented by Eq. 5
5: **return** (B^i, C^i)

cross-modality relationship between depth and pseudo-color maps. Since early fusion would lead to information lost due to both modalities has large gap in representation structure, we thus utilize idea of feature fusion instead of raw and early fusion. Then, defect detection module applies steps of region proposal, region classification and regression to accurately predict defect regions. Finally, post-processing module adopts morphological information of detected defects to perform alignment, thereby suppressing the wrong detection of non-overlapping areas.

We describe steps of the proposed defect detection algorithm in Algorithm 2. Specifically, each input batch of data can be represented as a set:$\{(S_d^1, S_c^1), (S_d^i, S_c^i), ..., (S_d^K, S_c^K)\}$, where i represents the index of splitting batch, K is the total batch number, S_d^i and S_c^i refer to the ith batch of depth and color imaging data, respectively. The later multi-modal feature extraction operation extracts semantic 3D and 2D features F_d^i and F_c^i based on input S_d^i and S_c^i, respectively. Then, feature fusion operation fuse F_d^i and F_c^i to obtain distinguished feature map F_f^i for further detection. After multiple modality feature extraction and fuse accomplished by Eq. 2 and Eq. 3 respectively, the defect detection result can be obtained by:

$$(B^i, C^i) = f_{det}(S_d^i, S_c^i), where\ 1 \leq i \leq K. \tag{1}$$

where function $f_{det}()$ refers to operations of the proposed defect detection module, B and C represent set of bounding boxes and the corresponding confidence scores as detection results.

Finally, post-processing operation is used to suppress wrong detection results based on Euclidean distance in feature map between $F_{f,j}^i$ and the pre-extracted defect prototype feature set $F_{f,p}$, which utilizes characteristics of general defects to improve robustness of small defect detection. It's worthy noticing that operations in algorithm are designed with sequential connections, where they are trained firstly in individual sections and then in a collaborative way, thus optimizing the whole process in first locally and then globally maximal sense.

3.4 Design of Multi-modal Feature Extraction and Fusion Module

Most of the existing image-based defect detection methods focus on extraction of information from single image modality, rather than multiple modalities. To boost detection performance even facing extremely small defects, we design to extract and fuse information from both modalities, i.e., the input depth and color imaging data. It's noted that this module is deployed on edge servers, transmitting fused feature map to cloud servers for further detection.

We perform feature extraction on color imaging data S_c through backbone network, i.e., Swin-T, which is built on self-attention network. Swin-T not only performs multi-level recursive feature extraction being similar with the classic convolutional neural network, but also constructs window-shifting self-attention scheme to perform multiple iterations of feature optimization. Moreover, Swin-T introduces a down-sampling operation similar to pooling, which is more conducive to expand size of receptive field. Owing to hierarchical structure of feature maps computed by Swin-T, end-to-end feature fusion methods like Feature Pyramid Fusion (FPN) can be directly applied.

Specifically, we first apply chunking operation to process raw color imaging data S_c, which is a common pre-processing step for feature extraction via transformer. More precisely, S_c is divided into non-overlapping sub-blocks with a stride of 4 pixels in both row and column directions. Each 3D feature map is constructed with multiple sub-blocks as $4 \times 4 \times 3 = 48$. Then, a fully connected layer is adopted to map dimension of the sub-block from 48 to ξ, where ξ is a preset constant. Afterwards, we use a window-shifting self-attention scheme for local feature extraction. Finally, we use three different sub-block fusion layers to down-sample the generated feature map, where the same self-attention scheme is further adopted to generate local feature with different scales. It's noted that sub-block fusion layer can reduce the number of sub-blocks to a quarter of the original, and double the dimension of sub-block, which is similar to pooling operation.

After feature extraction by Swin-T, we design pyramid fusion operation to fuse feature map of different scale, which is capable to effectively enhance feature representation ability, especially low-level ones to improve detection performance of small defects. Such operation can be written as:

$$\tilde{F}_{c,m} = f_{up}(\tilde{F}_{c,m-1}) \oplus f_{conv}(F_{c,m}), where\ 2 \leq m \leq 4 \tag{2}$$

where m refers to index number of pyramid level, $F_{c,m}$ and $\tilde{F}_{c,m}$ represent feature map before and after pyramid fusion operation respectively, $f_{up}()$ means up-sampling operation using nearest neighbor interpolation, $f_{conv}()$ represents using 1×1 convolution operation to reduce number of feature channels, and $\oplus$ means element-wise addition operation.

Similarly, the proposed module extract features from depth map data S_d with Swin-T and pyramid fusion operation, obtaining feature set $\{\tilde{F}_{d,m}\}, m = 1, 2, 3, 4$. It's noted $\tilde{F}_{d,m}$ for depth data and $\tilde{F}_{c,m}$ for color imaging data have the same size in different scales.

At last, we fuse $\tilde{F}_{d,m}$ and $\tilde{F}_{c,m}$ for information enhancement via multiple modalities, which can be written as

$$F_{f,m} = \omega_m \odot f_g(\tilde{F}_{c,m}) + (1 - \omega_m) \odot f_g(\tilde{F}_{d,m}), where\ 1 \leq m \leq 4, \tag{3}$$

where function $f_g()$ refers to global pooling operation for feature generation with dimension $1 \times D$, $\odot$ refers to element-wise multiplication, and ω_m is weight for different modality. It can be adaptively calculated based on input feature map of different modalities as:

$$\omega_m = f_{ful}(f_{con}(f_g(\tilde{F}_{c,m}), f_g(\tilde{F}_{d,m}))), where\ 1 \leq m \leq 4, \tag{4}$$

where $f_{con}()$ refers to concatenate operation, and $f_{ful}()$ represents two fully-connected layers. It's noted the number of nodes in each fully-connected layer is $2D$, $\frac{D}{4}$, and D respectively, and each layer uses ReLU and Sigmoid activation function respectively.

3.5 Designs of Defect Detection and Post-processing Module

In this subsection, we will introduce designs of defect detection and post-processing modules with three steps, i.e., ROI proposal, ROI classification and regression, and Post-processing via morphology operations.

ROI Proposal Step. In the first step, we adopt CNN to predict regions that may contain surface defects based on feature map computed by last module. Moreover, we adopt K-means algorithm to improve anchor box settings for higher accuracy.

Specifically, we firstly perform operation of generating region proposals on all levels of fused feature maps $\{F_{f,m}, m = 1, 2, 3, 4\}$, which encodes visual clues in different modalities and scales. Moreover, low-level feature map not only interacts with high-level feature map for semantical meaning boosting, but also involves local information for small defects, thus benefiting defect detection with high recall performance. Afterwards, we use K-means algorithm to cluster defect size based on the generated region proposals, thus setting size of the clustering centers as preset size of anchor boxes. In fact, K-means algorithm could largely promote to further classification step with an optimized initial values, thus achieving convergency in few iterations.

ROI Classification and Regression Step. Based on the preset size of anchor boxes, we not only classify defect categories and predict the exact bounding boxes by calibration on region proposals, but also offer prediction confidence for each group of prediction including category and bounding box. Therefore, we define B_j and C_j as the prediction of bounding box and confidence score for the jth defect located by intelligent detection algorithm f_{det}, where $1 \leq j \leq O$ and O is the total number of defects for the input and sensing product.

Post-processing Step via Morphology Operations After detection, there exist non-overlapping wrong detection results, due to weak deformation and similar patterns of defects. To suppress these errors for higher precision performance, a post-processing module is proposed, which performs morphological alignment by comparing between general patterns of defects and the current detected defect, thus suppressing non-usual defects by decreasing its prediction confidence.

Firstly, we collect quantity of typical defect samples to construct multi-modal feature set of defect prototypes F_p, which acts as a parametric conclusion on defeat patterns from depth and imaging modalities. Then, we scale all depth maps in F_p to the preset size $(200, 200)$ using nearest neighbor interpolation, and normalize them as values from 0 to 1.

Afterwards, we calculate distances $d_{j,n}$ in feature map between the jth bounding box $F_{f,j}$ and the nth defect prototype, where $1 \leq n \leq N$ and N is the total number of defect prototypes in training dataset. Once the minimal value in $d_{j,n}$ represented as $\tilde{d}_j$ is larger than a pre-set threshold δ, we would greatly decrease the corresponding prediction confidence C_j for suppressing and even eliminating. The whole process can be represented as follows:

$$C_j = \begin{cases} C_j, & if \;\; \tilde{d}_j \leq \delta \\ C_j - \frac{\tilde{d}_j}{2}, & if \;\; \tilde{d}_j > \delta \end{cases} \tag{5}$$

where $\tilde{d}_j$ is calculated as

$$\tilde{d}_j = \min \|F_{f,j} - F_{p,n}\|, where \; 1 \leq n \leq N, \tag{6}$$

where $\|\|$ refers to calculate Euclidean distance between two feature maps with same size.

4 EXPERIMENT

In this section, we show the effectiveness of the proposed DT framework in detecting small surface defeats. We first introduce dataset and measurements. Then, ablation experiments are conducted to prove positive impacts of different structure designs. Afterwards, we perform comparative studies on two novel modules and offer discussions on performance. Finally, we provide analysis on computation cost and implementation details.

4.1 Dataset and Measurement

We collect two datasets, i.e., DeA and DeB, from a factory, which corresponds to two industrial products A and B. Since occurrence probability of small surface defects is relatively low in all types of defects, we collect less samples than expected, where DeA and DeB contain 24 and 37 original 3D point clouds by scanning surface defeats of A and B, respectively. We show several examples in Fig. 4 by rendering 3D point clouds as pseudo color images for display, where we can observe rough surface, and complex texture appearance of A. Moreover, green rectangles are used to locate defects, which are difficult to recognize due to its small size and irregular shape. Essentially, all these properties reflect difficulties in real-world production scenario with DT sense, which improves generality of the proposed framework. DAGM 2007, KTH-TIPS, and several other datasets with the similar properties can be used for testing as well.

After obtaining DeA and DeB, we further construct DeA+ and DeB+, where original samples are manually annotated and enhanced to generate more samples. Specifically, we first use point cloud rendering algorithm,

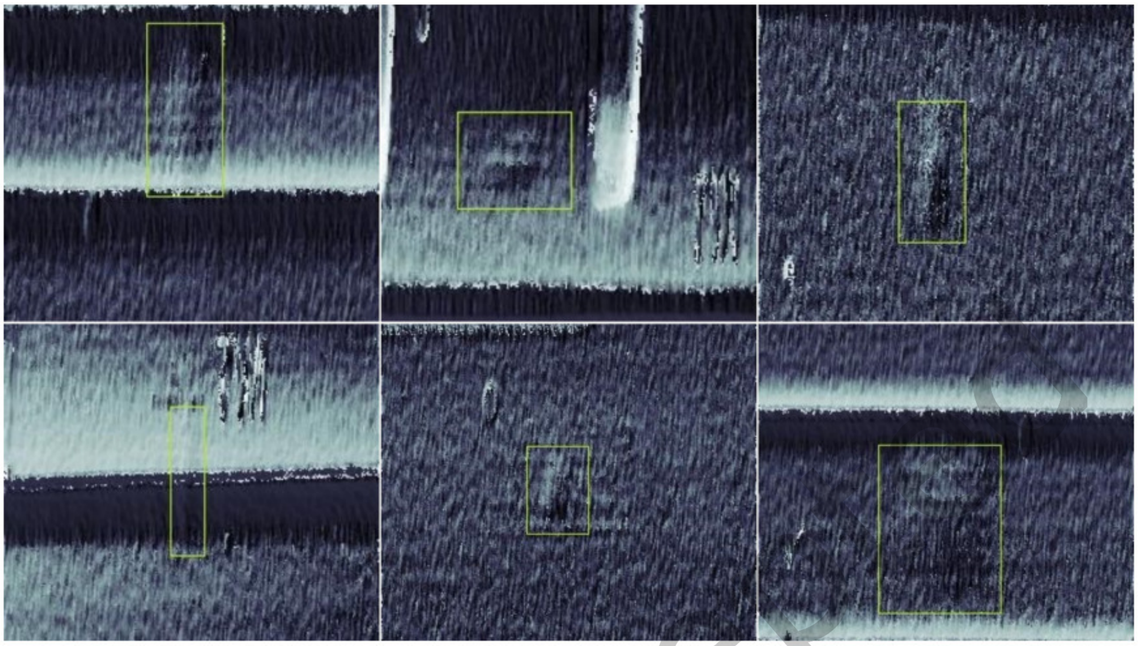


Fig. 4. Several samples of small surface defects in DeA dataset, where green rectangles refer to deformation defects.

which renders the original 3D point clouds as pseudo-color and depth map data with enhanced deformation characteristics. Then, we follow COCO annotation format for manual annotation based on pseudo-color map. To ensure fairness of testing during sample generation, we firstly divide original data into three parts by cross-validation criterion, and then generate another 300 samples in each part without interferences among testing and training samples. Finally, DeA+ and DeB+ is constructed by merging original and generated data, which can be represented as pairs of depth and color imaging data $S = \{(S_d^i, S_c^i), 1 \le i \le k\}$.

Following requirement analysis for algorithm design in Section 3.3, we apply AP, and recall for evaluation. Specifically, AP is defined as the mean precision value over multiple IoU (Intersection over Union) thresholds and all the object classes:

$$AP_{U_j} = \frac{1}{10 \times C} \sum_{i=1}^{C} \sum_{j=1}^{1} 0P(i, U_j) \tag{7}$$

where i and j refer to the index of class and threshold respectively, C is the total number of classes, the IoU values U_j corresponds to a range from 0.5 to 0.95 with a step size of 0.05, and the function $P(i, U_j)()$ calculates precision values for the ith object class under a fixed IoU threshold U_j. More precisely, AP_{50} refer to mAP values over the IoU thresholds of 0.5.

Recall is used to measure the capability of detection algorithm to accurately find out all defects from quantity of scanning products, where we expect to obtain high recall performance during testing. Since prediction results can be divided into four categories, i.e., true positive (TP), false negative (FN), false positive (FP), and true negative (TN), recall can be calculated with $Recall = N_{TP}/(N_{TP} + N_{FN})$, where N represents number of classified samples.

Table 1. Performance comparisons with different structure designs on DeA+ and DeB+ datasets. Bold indicates the best.

Dataset	Render	Enhance	Fusion	Post-pro	AP_{50}(%)	Recall(%)
DeA+	✓	✓	✓	✓	**75.2**	95.4
DeA+	✓	✓	✓	–	71.7	**96.9**
DeA+	✓	✓	–	–	71.2	89.7
DeA+	✓	–	–	–	67.3	82.4
DeA+	–	–	–	–	41.9	52.5
DeB+	✓	✓	✓	✓	**77.0**	97.7
DeB+	✓	✓	✓	–	72.4	**98.2**
DeB+	✓	✓	–	–	72.6	91.9
DeB+	✓	–	–	–	68.1	85.7
DeB+	–	–	–	–	45.9	64.7

4.2 Ablation Experiment

To explore effectiveness of structure designs, results of ablation experiments are shown in Table. 1, where we add algorithm modules on the basis of Faster RCNN network for performance comparisons. Specifically, **Post-pro** refers to add the proposed morphological alignment algorithm on the basis of NMS in the post-processing module. **Fusion** represents to adopt the proposed multi-modal feature extraction and fusion module, rather than only using one modality, i.e., pseudo color data, for training. **Enhance** refers to generate new samples based on morphological operations, rather than only adopting basic transformations for data enhancement, such as rotation, clipping, scaling and so on. **Render** represents that rendering the original 3D point clouds as pseudo-color and depth map data with enhanced deformation characteristics, rather than only using point clouds and depth maps for training.

From Table 1, we can observe that Render and Enhance settings greatly improve defect detection performance, proved by large promotion in AP_{50} and *Recall*. In fact, Render help generate 2D modality, i.e., pseudo-color data and offer more informative 3D modality, i.e., depth data on the basis of point cloud data, which offers multiple views to better locate small and weak defects as shown in Fig. 2. Meanwhile, Enhance greatly increases the number of samples in training dataset with novel morphological operations, which prevents overfitting of small dataset and improves generalization ability of the trained network.

Later, we could observe that Fusion could greatly increase recall performance, but fails in promoting AP_{50}. This phenomenon can be explained by the fact that fusion introduces multiple modalities with new visual clues on defects, which helps to mine all possible defects with a high recall performance. However, new possible defects are difficult to accurately locate due to its small and weak deformation properties, resulting in the same or even lower AP_{50} performance.

Last but not least, Post-pro greatly improves AP_{50}, meanwhile slightly decreasing recall performance. Essentially, post-processing operations, including NMS and the proposed morphological alignment algorithm, generally help suppress non-overlapping false detection defects, thus increasing AP_{50} and decreasing recall explained by definitions of both measurements.

Table 2. Performance comparisons between the proposed method and several comparative studies on DeA+ and DeB+ datasets. It's noted that we modify settings of multimodal feature extraction and fusion module for comparisons. Bold indicates the best.

Dataset	Backbone	Fusion	AP_{50}(%)	Recall(%)
DeA+	ResNet18	OnlyColor	69.5	85.5
DeA+	ResNet18	OnlyDepth	65.2	81.3
DeA+	ResNet18	FuseAdd	71.1	88.3
DeA+	ResNet18	FuseConcat	69.8	90.4
DeA+	ResNet18	Ours	**72.1**	**94.3**
DeA+	ResNet50	OnlyColor	70.0	88.3
DeA+	ResNet50	OnlyDepth	64.9	80.8
DeA+	ResNet50	FuseAdd	70.8	90.1
DeA+	ResNet50	FuseConcat	71.3	91.3
DeA+	ResNet50	Ours	**71.7**	**96.9**
DeB+	ResNet18	OnlyColor	70.8	89.5
DeB+	ResNet18	OnlyDepth	68.5	84.4
DeB+	ResNet18	FuseAdd	70.2	93.1
DeB+	ResNet18	FuseConcat	**71.8**	92.1
DeB+	ResNet18	Ours	**71.8**	**97.1**
DeB+	ResNet50	OnlyColor	71.8	90.2
DeB+	ResNet50	OnlyDepth	68.7	85.2
DeB+	ResNet50	FuseAdd	72.1	94.6
DeB+	ResNet50	FuseConcat	**72.7**	94.1
DeB+	ResNet50	Ours	72.4	**98.2**

Based on all former analysis, it's our best choice to adopt all four modules for the highest AP_{50} and second highest recall, which keeps balance between precision and recall measurement, thus promoting intelligent and applicable capability of the whole framework.

4.3 Comparative Experiment on Multimodal Feature Extraction and Fusion Module

We report defect detection results achieved by the proposed method and several comparative methods in Table. 2, where we modify settings of multimodal feature extraction and fusion module for comparisons. Specifically, **OnlyColor** abandons structure of data fusion with only pseudo-color data. On the contrary, **OnlyDepth** abandons structure of data fusion with only depth data. **FuseAdd** adopts feature fusion method with element-wise addition operation, where feature map of both modalities directly sums for output. **FuseConcat** use concatenation operation and 1×1 convolution filter for feature fusion, where feature map of both modalities are firstly concatenated as one feature map and then re-scaled by convolution operation.

On both DeA+ and DeB+ datasets, the proposed method generally achieves the best performance in terms of AP_{50} and recall, except that we achieve slightly worse performance than FuseConcat in group of DeB+ and ResNet50. It's noted that the proposed method has achieved large improvement in recall, since design of multi-modal feature fusion enables to better locate small and weak deformation defects by viewing and examining surface patterns via distinguish feature maps. Adopting one modality, such as OnlyColor and OnlyDepth, fails

Table 3. Performance comparisons between the proposed method and several comparative studies on DeA+ and DeB+ datasets. It's noted that we modify settings of post-processing module for comparisons. Bold indicates the best.

Dataset	Backbone	Post-pro	AP_{50}(%)	Recall(%)
DeA+	ResNet18	NMS	72.1	**94.3**
DeA+	ResNet18	NMS+Ours	**76.8**	93.6
DeA+	ResNet18	softNMS	70.1	**94.3**
DeA+	ResNet18	softNMS+Ours	74.6	93.6
DeA+	ResNet50	NMS	71.7	**96.9**
DeA+	ResNet50	NMS+Ours	**75.2**	95.4
DeA+	ResNet50	softNMS	70.9	**96.9**
DeA+	ResNet50	softNMS+Ours	73.6	95.4
DeB+	ResNet18	NMS	71.8	**97.1**
DeB+	ResNet18	NMS+Ours	**74.7**	96.5
DeB+	ResNet18	softNMS	70.2	**97.1**
DeB+	ResNet18	softNMS+Ours	73.3	96.5
DeB+	ResNet50	NMS	72.4	**98.2**
DeB+	ResNet50	NMS+Ours	**77.0**	97.7
DeB+	ResNet50	softNMS	71.6	**98.2**
DeB+	ResNet50	softNMS+Ours	74.2	97.7

to search for the best view to locate defects without abundant information, which is proved by the fact that their results are quite smaller than the other three methods. Moreover, the proposed adaptive weighting scheme offers weights on feature maps of different modalities based on input content information, thus achieving more convinced and accurate detection results. Such advantage can be proved by the fact that the proposed method outperforms FuseAdd and FuseContact in all testings, which apply fixed and inflexible fusion strategy for multiple modalities fusing.

The proposed method has a smaller gain on AP_{50}, compared with recall measurement. This phenomenon can be explained by the fact that adopting multiple modalities offers more potential defect regions to improve recall, nevertheless bringing difficulties in identifying them as defects or not with their complicated input raw data. We further find these hard cases as non-overlapping bounding boxes, where algorithm misclassify them due to their similar appearance and texture with defects. To distingue such hard cases for precision boosting, we thus design post-processing module with idea of morphological alignment.

Experimental results also show that ResNet50 is more proper than ResNet18 to be obtained for backbone network for defect detection, where deeper structure of ResNet50 is capable to extract more informative and fine-grained features for locating small defects, compared with relatively shallow network depth of ResNet18.

4.4 Comparative Experiment on Post-processing Module

Table 3 shows comparative experimental results on DeA+ and DeB+ datasets, where we modify settings of post-processing module as comparative studies. Specifically, **NMS** sorts detection bounding boxes of the same category based on their corresponding confidence scores, thus eliminating boxes with larger IoU than threshold. Meanwhile, **SoftNMS** removes detection bounding boxes whose confidence are smaller than threshold by decreasing confidence scores based on their IoU values. It's noted that all post-processing algorithms in this paper

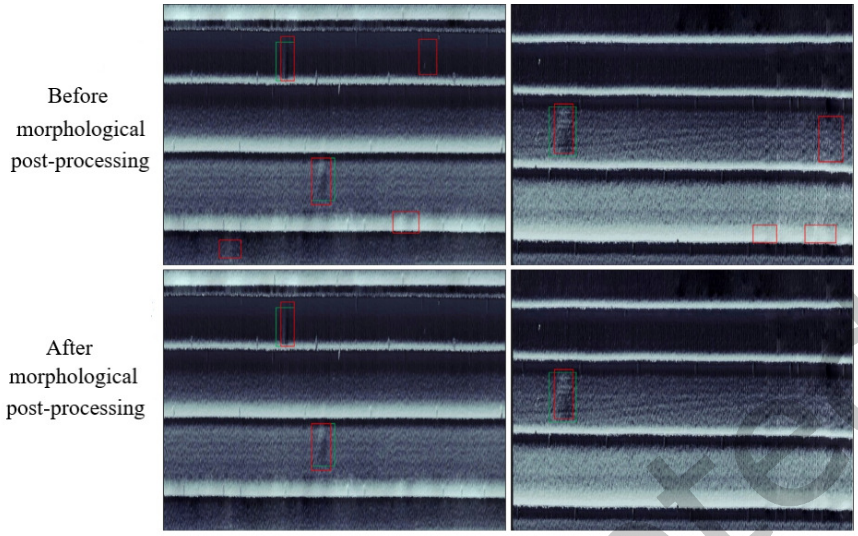


Fig. 5. Samples of the detected defeats before and after morphological post-processing operations.

are designed without learning process, so that they can be merged in sequential order for accuracy boosting. We show samples of the detected defeats before and after morphological post-processing operations in Fig.5, where we can observe that proper post-processing algorithm could greatly prevent error detections, even with similarities in appearance and texture.

It's observed that the proposed morphological alignment algorithm improves AP_{50} and slightly decreases recall on the basis of NMS and softNMS. Such experimental results show that morphological post-processing can effectively suppress non-overlapping false detection regions to improve precision performance. Meanwhile, NMS or softNMS is arranged in sequential processing order to deal with overlapping false detections. However, the proposed morphological alignment algorithm eliminates a small number of correct detection regions, since their shape patterns are not included in the pre-extracted prototype dataset. In fact, this drawback can be avoided by enlarging size of prototype dataset with more captured samples.

Only using NMS or softNMS achieves the same recall and different AP_{50} as illuminated in Table 3, since post-processing methods help remove wrong detections other than find more defeats. Moreover, NMS generally achieves better AP_{50} performance than softNMS, no matter using only or with the morphological alignment algorithm. This phenomenon can be explained by the fact that softNMS achieves more redundant bounding boxes on sparse data, due to its strategy to suppress wrong detections via decay in confidence score. Last but not least, usage of ResNet50 promotes to defeat detection performance due to its deeper network structure, compared with shallow structure of ResNet18.

4.5 Computation Cost

In this subsection, we only discuss the time-consuming part of our surface defect detection method, including transmission time, defect collection time and processing time in both edge servers and cloud center. Under the hardware and software environment mentioned in Section 4.6, the transmission of 1833ąÁ1396ąÁ1 image mentioned in DeA+ is total cost 2.55s between the edge server and cloud center, which is the same as the transmission time between the devices and edge servers. The simulation results show that the processing time of each image in edge servers is 9.76s, and the processing time in cloud center is 8.74s. After all, the proposed method is still applicable to the actual surface defect detection scene, and simulation is only a means to verify the effectiveness and correctness of the proposed surface defect detection system.

4.6 Implementation Details

All our experiments were conducted on a server with two Intel Xeon E5-2620 v4 (@2.1GHz) CPUs and one single NVIDIA GTX 1080Ti graphics cards. We adopt three-fold cross-validation to divide training and testing set. ImageNet dataset is used to pre-train weights. The training learning rate and batch size is set to 0.001 and 1, respectively. All methods in comparative experiments are trained for 50 epochs. To evaluate the accurate computation cost of the cloud-edge structure, we choose the Amazon's reserved instance "m3.medium" as the virtual machines (VM) on the edge servers.

5 CONCLUSION

Automatical defect detection is widely used in manufacturing. However, it's still difficult to construct relationship between twin simulation and real scenario considering dynamic variations, especially when dealing with small surface defects. We thus propose a framework of intelligent small surface defect detection with CMS technologies for DT, including an Edge-Cloud architecture and an intelligent surface defect detection algorithm. Considering dynamic characteristics and real-time response requirement, Edge-Cloud architecture is built to efficiently collect, process, analyze, and store big data produced by stream lines of factory. Then, we extract and fuse features from both 2D and 3D modalities to accurately identify the status of surface. Finally, a novel morphological alignment algorithm is proposed to aid in eliminating wrong detection for precision boosting. Ablation and comparative experiments prove the effectiveness of the proposed method in building DT environment for small defeat detection. Our future work includes 3D geometry reconstruction via multi-view captured images to promote detection accuracy with surface geometry information.

6 ACKNOWLEDGE

This work was supported by National Key R&D Program of China under Grant No. 2021YFB3900601, National Natural Science Foundation of China under Grant No. 62172438, the Fundamental Research Funds for the Central Universities under Grant B220202074, and the Fundamental Research Funds for the Central Universities, JLU.

REFERENCES

[1] Samah Aloufi and Abdulmotaleb El-Saddik. 2022. MMSUM Digital Twins: A Multi-view Multi-modality Summarization Framework for Sporting Events. *ACM Trans. Multim. Comput. Commun. Appl.* 18, 1 (2022), 5:1–5:25.

[2] Peter Bauer, Bjorn Stevens, and Wilco Hazeleger. 2021. A digital twin of Earth for the green transition. *Nature Climate Change* 11, 2 (2021), 80–83.

[3] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. 2017. Soft-NMS–improving object detection with one line of code. In *Proceedings of the IEEE international conference on computer vision*. 5561–5569.

[4] Chen Chen, Jiange Jiang, Yang Zhou, Ning Lv, Xiaoxu Liang, and Shaohua Wan. 2022. An edge intelligence empowered flooding process prediction using Internet of things in smart city. *J. Parallel and Distrib. Comput.* 165 (2022), 66–78.

[5] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 2016. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2147–2156.

[6] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. 2017. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 1907–1915.

[7] Yuwei Cheng, Hu Xu, and Yimin Liu. 2021. Robust Small Object Detection on the Water Surface through Fusion of Camera and Millimeter Wave Radar. In *Proceedings of IEEE/CVF International Conference on Computer Vision*. IEEE, 15243–15252.

[8] Songtao Ding, Hongyu Wang, Hu Lu, Michele Nappi, and Shaohua Wan. 2022. Two path gland segmentation algorithm of colon pathological image based on local semantic guidance. *IEEE Journal of Biomedical and Health Informatics* (2022).

[9] Martin Engelcke, Dushyant Rao, Dominic Zeng Wang, Chi Hay Tong, and Ingmar Posner. 2017. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1355–1361.
[10] Shahrzad Faghih-Roohi, Siamak Hajizadeh, Alfredo Núñez, Robert Babuska, and Bart De Schutter. 2016. Deep convolutional neural networks for detection of rail surface defects. In *Proceedings of International Joint Conference on Neural Networks*. 2584–2589.
[11] Liuhao Ge, Yujun Cai, Junwu Weng, and Junsong Yuan. 2018. Hand pointnet: 3d hand pose estimation using point sets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8417–8426.
[12] Ross Girshick. 2015. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 1440–1448.
[13] Yu He, Kechen Song, Qinggang Meng, and Yunhui Yan. 2019. An end-to-end steel surface defect detection approach via fusing multiple hierarchical features. *IEEE Transactions on Instrumentation and Measurement* 69, 4 (2019), 1493–1504.
[14] Yihui He, Chenchen Zhu, Jianren Wang, Marios Savvides, and Xiangyu Zhang. 2019. Bounding box regression with uncertainty for accurate object detection. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*. 2888–2897.
[15] Sabina Jeschke, Christian Brecher, Tobias Meisen, Denis Özdemir, and Tim Eschert. 2017. Industrial internet of things and cyber manufacturing systems. In *Industrial internet of things*. Springer, 3–19.
[16] Xiaogang Jia, Xianqiang Yang, Xinghu Yu, and Huijun Gao. 2020. A Modified CenterNet for Crack Detection of Sanitary Ceramics. In *Proceedings of The Annual Conference of the IEEE Industrial Electronics Society*. 5311–5316.
[17] Eric Guiffo Kaigom and Jürgen Roßmann. 2021. Value-Driven Robotic Digital Twins in Cyber-Physical Applications. *IEEE Trans. Ind. Informatics* 17, 5 (2021), 3609–3619.
[18] Hei Law and Jia Deng. 2020. CornerNet: Detecting Objects as Paired Keypoints. *Int. J. Comput. Vis.* 128, 3 (2020), 642–656.
[19] Jay Lee, Behrad Bagheri, and Chao Jin. 2016. Introduction to cyber manufacturing. *Manufacturing Letters* 8 (2016), 11–15.
[20] Xiaoming Li, Hao Liu, Weixi Wang, Ye Zheng, Haibin Lv, and Zhihan Lv. 2022. Big data analysis of the Internet of Things in the digital twins of smart city based on deep learning. *Future Gener. Comput. Syst.* 128 (2022), 167–177.
[21] Hui Lin, Bin Li, Xinggang Wang, Yufeng Shu, and Shuanglong Niu. 2019. Automated defect inspection of LED chip using deep convolutional neural network. *J. Intell. Manuf.* 30, 6 (2019), 2525–2534.
[22] Hsien-I Lin and Fauzy Satrio Wibowo. 2021. Image Data Assessment Approach for Deep Learning-Based Metal Surface Defect-Detection Systems. *IEEE Access* 9 (2021), 47621–47638.
[23] Su Liu, Jiong Yu, Xiaoheng Deng, and Shaohua Wan. 2021. FedCPF: An efficient-communication federated learning approach for vehicular edge computing in 6G communication networks. *IEEE Transactions on Intelligent Transportation Systems* 23, 2 (2021), 1616–1629.
[24] László Monostori, Botond Kádár, Thomas Bauernhansl, Shinsuke Kondoh, Soundar Kumara, Gunther Reinhart, Olaf Sauer, Gunther Schuh, Wilfried Sihn, and Kenichi Ueda. 2016. Cyber-physical systems in manufacturing. *Cirp Annals* 65, 2 (2016), 621–641.
[25] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 2017. 3d bounding box estimation using deep learning and geometry. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 7074–7082.
[26] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 652–660.
[27] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems* 30 (2017).
[28] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems* 30 (2017).
[29] Domen Racki, Dejan Tomazevic, and Danijel Skocaj. 2018. A Compact Convolutional Neural Network for Textured Surface Anomaly Detection. In *Proceedings of IEEE Winter Conference on Applications of Computer Vision*. 1331–1339.
[30] Roberto Saracco. 2019. Digital Twins: Bridging Physical Space and Cyberspace. *Computer* 52, 12 (2019), 58–64.
[31] Michael Schluse, Marc Priggemeyer, Linus Atorf, and Jürgen Rossmann. 2018. Experimentable Digital Twins - Streamlining Simulation-Based Systems Engineering for Industry 4.0. *IEEE Trans. Ind. Informatics* 14, 4 (2018), 1722–1731. https://doi.org/10.1109/TII.2018.2804917
[32] Wen Sun, Ning Xu, Lu Wang, Haibin Zhang, and Yan Zhang. 2022. Dynamic Digital Twin and Federated Learning With Incentives for Air-Ground Networks. *IEEE Trans. Netw. Sci. Eng.* 9, 1 (2022), 321–333.
[33] Domen Tabernik, Samo Sela, Jure Skvarc, and Danijel Skocaj. 2020. Segmentation-based deep-learning approach for surface-defect detection. *J. Intell. Manuf.* 31, 3 (2020), 759–776.
[34] Dazhong Wu, Shaopeng Liu, Li Zhang, Janis Terpenny, Robert X. Gao, Thomas Kurfess, and Judith A. Guzzo. 2017. A fog computing-based framework for process monitoring and prognosis in cyber-manufacturing. *Journal of Manufacturing Systems* 43 (2017), 25–34.
[35] M. Wu and Y. Moon. 2018. Intrusion Detection for CyberManufacturing System. *Journal of Manufacturing Science and Engineering* (2018).
[36] Yirui Wu, Haifeng Guo, Chinmay Chakraborty, Mohammad Khosravi, Stefano Berretti, and Shaohua Wan. 2022. Edge computing driven low-light image dynamic enhancement for object detection. *IEEE Transactions on Network Science and Engineering* (2022).
[37] Yan Yan, Yuxing Mao, and Bo Li. 2018. Second: Sparsely embedded convolutional detection. *Sensors* 18, 10 (2018), 3337.

[38] Chao Zhang, Guanghui Zhou, Han Li, and Yan Cao. 2020. Manufacturing Blockchain of Things for the Configuration of a Data- and Knowledge-Driven Digital Twin Manufacturing Cell. *IEEE Internet Things J.* 7, 12 (2020), 11884–11894.
[39] Liang Zhao, Chengcheng Wang, Kanglian Zhao, Daniele Tarchi, Shaohua Wan, and Neeraj Kumar. 2022. INTERLINK: A digital twin-assisted storage strategy for satellite-terrestrial networks. *IEEE Trans. Aerospace Electron. Systems* (2022).
[40] Xiaochi Zhou, Andreas Eibeck, Mei Qi Lim, Nenad B Krdzavac, and Markus Kraft. 2019. An agent composition framework for the J-Park Simulator-A knowledge graph for the process industry. *Computers & Chemical Engineering* 130 (2019), 106577.
[41] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. 2019. Objects as Points. *CoRR* abs/1904.07850 (2019).
[42] Xiaokang Zhou, Xuesong Xu, Wei Liang, Zhi Zeng, Shohei Shimizu, Laurence T. Yang, and Qun Jin. 2022. Intelligent Small Object Detection for Digital Twin in Smart Manufacturing With Industrial Cyber-Physical Systems. *IEEE Trans. Ind. Informatics* 18, 2 (2022), 1377–1386.