IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 8, 2021, accepted September 20, 2021, date of publication September 29, 2021,
date of current version October 8, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3116799

Bi-Directional Convolutional Recurrent Reconstructive Network for Welding Defect Detection

YOUNG-MIN KIM[1], IN-UG YOON[2], HYUN MYUNG[2], (Senior Member, IEEE), AND JONG-HWAN KIM[2], (Fellow, IEEE)

[1]Robotics Program, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Republic of Korea
[2]School of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Republic of Korea

Corresponding author: Jong-Hwan Kim (johkim@rit.kaist.ac.kr)

This work was supported by DSEC, a marine engineering company in Korea and partially by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by Korea government Ministry of Science and ICT (MSIT) under Grant No. 2020-0-00440 (Development of Artificial Intelligence Technology that Continuously Improves Itself as the Situation Changes in the Real World).

ABSTRACT Nowadays, the welding process is essential in various manufacturing industrial fields, such as aerospace, vehicle production, and shipbuilding. The welding defects caused in the process need to be monitored as they can cause serious accidents and losses. Traditional computer vision methods in an industrial application are inefficient when the detection targets have variations in shape, scale, and color because the detection performance depends on the hand-crafted features. To overcome this limitation, deep learning models, such as the convolutional neural network (CNN), are applied to industrial defect detection. These CNN-based models trained on static images, however, have a low performance that cannot meet the industrial requirements. To deal with the challenge, bidirectional Convolutional Recurrent Reconstructive Network (bi-CRRN) is proposed for welding defect detection and localization based on welding video. Spatio-temporal data, specifically the forward and backward sequences, are considered in our bi-CRRN to get high detection performance. Moreover, an automatic defect detection equipment is developed to weld a material and monitor the welding bead simultaneously. We demonstrate that the proposed bi-CRRN outperforms the other segmentation network models in welding defect detection.

INDEX TERMS Convolutional recurrent reconstructive network (CRRN), bi-CRRN, convolutional LSTM, spatiotemporal data, defect detection.

I. INTRODUCTION

Deep learning has shown significant progress in various fields, including image classification, semantic segmentation, and object detection. In industrial applications, these advanced algorithms have led to a dramatic increase in performance resulting in the improvement in productivity and efficiency. One of the major improvements is achieved in the defect detection field. The defect detection has been a daunting task for the engineers due to the high accuracy demands, periodical examination, and immense examination areas. To deal with such challenges, an automated defect detection system has been developed using a deep learning based approach. It could reduce the human labor and enhance the accuracy and efficiency. In the welding inspection, the automation based on deep learning algorithms invokes an increase in the reliability and reproducibility of the task. Furthermore, it speeds up the process and decreases labor costs and human errors.

The automated defect detection system employs an acquisition equipment to obtain images that are used as the defect detection input. This acquisition equipment contains various measurement devices such as the RGB camera, depth camera, and ultrasonic devices. RGB camera is most widely used owing to its similarity with the human visual inspection. Furthermore, RGB camera-based systems achieve high accuracy and provide an intuitive understanding of images during the process [1], [2].

After the acquisition of RGB images, the defect detection algorithm highlights the defected area within the images. The algorithm is classified into image-wise and pixel-wise methods. The image-wise defect detection method determines

The associate editor coordinating the review of this manuscript and approving it for publication was Jiankang Zhang.

the existence of a defect within the entire image [3], while the pixel-wise defect detection method determines the specific defect locations in pixel-level [4], [5]. The advantage of the image-wise defect detection method is the reduced network size. The latent-to-image decoder is not necessary, thus designing the network is less complex as compared to the pixel-wise method. On the other hand, the pixel-wise defect detection not only indicates the presence of a defect but also the location of the defect. Knowing the location of the defect can help optimize the process of the industrial field line. In addition, the pixel-wise defect detection result is an important factor in evaluating the quality of the product. In this light, the pixel-wise method is generally preferred in the industrial defect detection problem, where the location of product defects is required.

However, to utilize the pixel-wise technique on the industrial level, the following three issues need to be resolved. Firstly, the network size is too large so that it requires more inference time and limits the real-time detection. Secondly, spatial information needs to be preserved and employed throughout the network architecture. Lastly, to achieve a higher prediction score in a harsh industrial field, which is difficult to attain static images, the time-sequential information should also be utilized.

The recent line of works has been attempting to deal with such issues. Networks considering spatio- or spatio-temporal information within images have been developed [6], [7]. Furthermore, efforts to reduce the network size while maintaining the high performance have also been made [8]. Most of the researches, however, implemented unsupervised learning architecture [9], [10] only and did not target on defect detection. Traditionally, methods based on the unsupervised learning algorithm typically provide lower performance as compared to the supervised learning algorithm.

In this paper, we propose the bidirectional convolutional recurrent reconstructive network (bi-CRRN) for real-time pixel-wise defect detection, which utilizes spatio-temporal information in videos. Three major contributions are presented here. Firstly, we develop an automatic defect detection equipment to obtain videos as input, and detect the welding defects. The equipment also includes a setup for acquiring the training data manually. Secondly, we design the bi-CRRN algorithm to utilize the spatio-temporal information from the relationship between input images in both forward and backward directions, by adopting the bi-directional LSTM [11] structure. Finally, we compare the performance of the proposed bi-CRRN with recent defect detection algorithms on acquired welding datasets in terms of the accuracy at both frame and pixel levels along with computation time. Evidently, the proposed bi-CRRN outperforms both defect detection accuracy and computation speed.

This paper is organized as follows. In Section II, we describe related works including automated defect detection systems and spatio-temporal networks. Section III briefly reviews the mechanism of CRRN and presents the proposed bi-CRRN followed by the experimental validation in Section IV. Finally, concluding remarks follow in Section V.

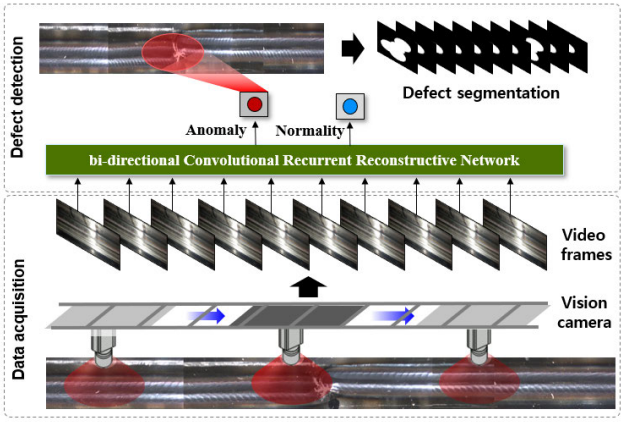

FIGURE 1. Overall framework of the welding defect detection equipment. The vision camera attached to the welding machine takes a video of the welding site. Video frames are applied to the bi-CRRN for defect detection. Trained bi-CRRN determines the existence of the defect on the welding site.

II. RELATED WORKS

A. AUTOMATED DEFECT DETECTION SYSTEMS

In the industrial sites, the presence of defects in manufacturing products can cause several losses such as degradation in production quality, exposure to dangerous materials, and even catastrophic accidents. Various researches have been conducted for defect detection to prevent the losses caused by defects. Traditional methods for defect detection are manual inspections by highly trained human experts. These methods, however, require high labor costs and are highly prone to human errors due to inattention [12]. Thus, automation of defect detection has been widely studied to reduce these errors and operation costs.

An automated defect detection system requires various sensory equipment such as vision cameras, ultrasonic sensors, and radar sensors. The vision camera based imaging system is widely used due to its high performance and similarity to the human visual inspection [13]–[15]. However, traditional vision-based defect detection algorithm suffers from a performance robustness issue. The performance of these algorithms is highly volatile and can be easily affected by small changes in image features such as illuminance, scale variation, or object shape.

B. DEEP LEARNING BASED DEFECT DETECTION

Recent researches attempt to overcome the difficulties mentioned above by developing various networks with diverse characteristics. The most basic network for image processing is to utilize a CNN [16], due to its high computational efficiency and preservation of spatial information. Semantic segmentation classifies the image to objects pixel-wise, but the network is excessively bulky. Class activation mapping (CAM) based on CNN [17] provides spatial reasoning for classification results. However, it does not consider a

spatio-temporal relationship between input images. Networks for video inputs need to consider spatio-temporal characteristics of inputs for higher accuracy and improved efficiency.

In recent years, several other machine learning techniques have been applied to many industrial applications for robust performance [18]–[20]. In the case of railways, an automated rail defect detection system was studied, using a deep convolutional neural network (DCNN) [21]. Cha *et al.* [22] cropped the building surface images into patches and detected defects with the help of the CNN. Hu *et al.* [23] implemented defect detection on radiography images using bilinear class activation maps (Bi-CAM) and attention mechanisms. Kang *et al.* [24] proceeded with high-speed railway insulator defect detection utilizing faster R-CNN [25].

Likewise, automated visual inspections with machine learning techniques have also been applied to welding defect detection. Welding is an essential and commonly used technique in various mechanical industrial fields, including automobiles, aerospace, and shipbuilding. Hence, there have been diverse researches on the automation of welding defect detection. Lee *et al.* [26] showed that the artificial neural network (ANN) offered better prediction performance than using multiple regression analysis on back-bead prediction in gas metal arc welding (GMAW). Feng *et al.* [27] utilized an ensemble model incorporating multiple object detection networks for gas tungsten arc welding (GTAW) defect detection. Sassi *et al.* [28] monitored the welding defects in fuel injectors using transfer learning.

C. NETWORKS CONSIDERING SPATIO-TEMPORAL INFORMATION

Further researches have been focusing on the development of spatio-temporal pixel-wise network. Convolutional LSTM (ConvLSTM) network [6] preserves spatial information and considers the relationship between input images by applying convolutional operators to LSTM-based structure. Spatio-Temporal LSTM (ST-LSTM) [7] is designed to facilitate the flows of the spatio-temporal information by adding a spatio-temporal memory cell. In addition to the spatio-temporal memory in which the memory cell is updated in the time domain, a memory structure is also added to ST-LSTM which updates vertically for each layer within the same time step. Thus, it requires twice as many parameters as the ConvLSTM. Convolutional Recurrent Reconstructive Network (CRRN) [8] simplifies the network and reduces the necessary amount of the network parameters while maintaining the performance similar to the ST-LSTM. CRRN is utilized as an anomaly detection algorithm based on unsupervised learning. In the case of the industrial defect detection problem, however, supervised learning models tend to be more accurate than unsupervised learning models.

III. PROPOSED APPROACH

A. WELDING DEFECT DETECTION FRAMEWORK

The proposed welding defect detection framework is shown in Fig. 1, which is divided into three phases: 1) Automatic welding and obtaining input videos simultaneously; 2) Defect detection and localization at the pixel-level by applying a deep learning network; 3) Defect detection at the frame-level on the basis of pixel-level detection.

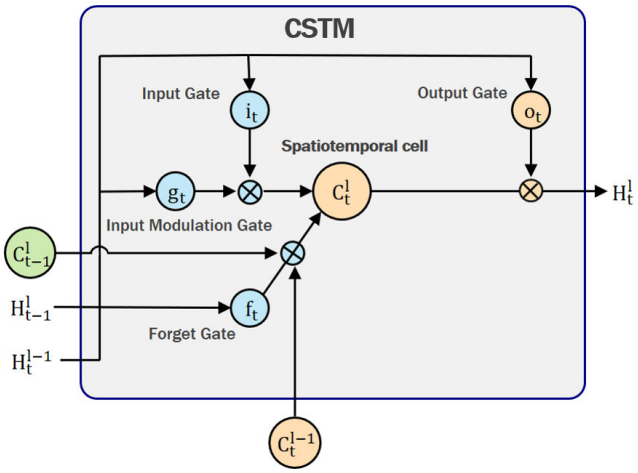

FIGURE 2. The Convolutional Spatio-Temporal Memory (CSTM) architecture, designed to enhance computational speed and reduce required memory amounts while maintaining the performance.

Firstly, when the automatic welding machine proceeds with automated welding, the welding bead is captured by the automatic welding defect detection system installed behind the welding machine. To improve the processing speed of the network, images are resized to smaller dimensions. Secondly, the deep learning network detects and localizes defects at the pixel-level. For enhanced defect detection performance, we design two bi-CRRN models. Lastly, the input images are classified into defective and normal classes. If the input image is classified to be defective, a speaker attached to the equipment generates an alarm signal to notify the operator about the possible defects. The predicted outputs with possible defects are saved automatically to a computer with corresponding input images.

B. CRRN

The automatic welding defect detection is performed based on the RGB vision camera. Since images are continuously captured over time, they contain not only spatial information of the welding bead but also temporal information. Therefore, we adopt CRRN [8] as a basic architecture, which is a convolutional recurrent autoencoder based on Convolutional Spatio-Temporal Memory (CSTM) for the anomaly detection in spatio-temporal data. The CRRN is a combination of encoder-decoder consisting of a spatial encoder (S-Encoder), a spatio-temporal encoder-decoder (ST-Encoder-Decoder), and a spatial decoder (S-Decoder). The S-Encoder extracts spatial features from the input, and the ST-Encoder extracts spatio-temporal features from a sequence of the spatial features. In a similar manner to the encoder, the ST-Decoder decodes spatial features of each timestep and the S-Decoder generates reconstructed outputs. These outputs help to

measure the reconstructed error between the input and the reconstructed output. From the reconstructed error, it is determined whether the input is normal or abnormal.

To extract spatio-temporal patterns efficiently in CRRN, a CSTM is developed to deliver spatial information to other CSTMs without incrementing the number of parameters, as shown in Fig. 2. In CSTM, firstly, the cell gate of the previous time step and the previous layer are concatenated in a channel-wise manner. Then, the cell is updated by adjusting the number of existing channels through one by one convolution. Therefore, fewer parameters compared to ST-LSTM [7] are used and both spatial and temporal patterns can be extracted. The CSTM is updated as follows:

$$\begin{aligned} g_t^l &= tanh(W_g * H_t^{l-1} + U_c * H_{t-1}^l) \\ i_t^l &= \sigma(W_i * H_t^{l-1} + U_i * H_{t-1}^l + W_c \circ C_{t-1}^l) \\ f_t^l &= \sigma(W_f * H_t^{l-1} + U_f * H_{t-1}^l + W_f \circ C_{t-1}^l) \\ C_t^l &= f_t^l \circ W_{1\times1}[C_{t-1}^l; C_t^{l-1}] + i_t^l \circ g_t^l \\ o_t^l &= \sigma(W_o * H_{t-1}^l + U_o * H_{t-1}^l + W_o \circ C_t^l) \\ H_t^l &= o_t^l \circ tanh(C_t^l), \end{aligned} \tag{1}$$

where * and ∘ denote a convolution operator and Hadamard product, respectively. Subscript l and t denote a layer and time step, respectively. g_t^l, i_t^l, f_t^l, and o_t^l denote an input modulation gate, input gate, forget gate, and output gate, respectively. C, H, U, and W are memory cell, hidden state, learnable weight for previous layer, and learnable weight for current layer, respectively. Two cell gates C_{t-1}^l and C_t^{l-1} are concatenated in a channel-wise manner. $W_{1\times1} \in \mathbb{R}^{N_c \times 2N_c}$ represents one by one convolutional operation weight matrix.

C. SUPERVISED bi-CRRN FRAMEWORK

In bi-CRRN, the S-Encoder extracts the spatial feature of the image. Then, the spatial and temporal information is processed by the ST-Encoder, which is composed of the CSTM modules. Next, the ST-Decoder exploits spatial and temporal patterns if it is used as a component of the network. Then, the processed information is passed to the S-Decoder, which predicts the final output.

To encode and decode the sequential information, only the temporal forward direction is considered in CRRN. Yet, the detection performance would be enhanced if both the directions of the sequential information are considered. Taking advantage of this presumption, we design two kinds of bi-CRRN, bi-CRRN-E and bi-CRRN-ED, both capable of processing forward and backward time sequences. To prevent decreasing the defect detection performance, bi-CRRN is trained by the supervised learning framework. The supervised learning method tends to be more accurate than the unsupervised one in general. The labeled data, which is addressed in the supervised learning as the target, is prepared by the welding expert at the pixel-level.

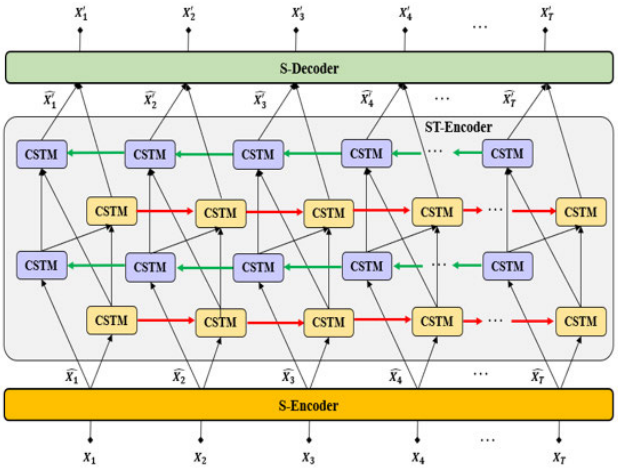

FIGURE 3. The bi-CRRN-E architecture using S-Encoder,S-Decoder and ST-Encoder.

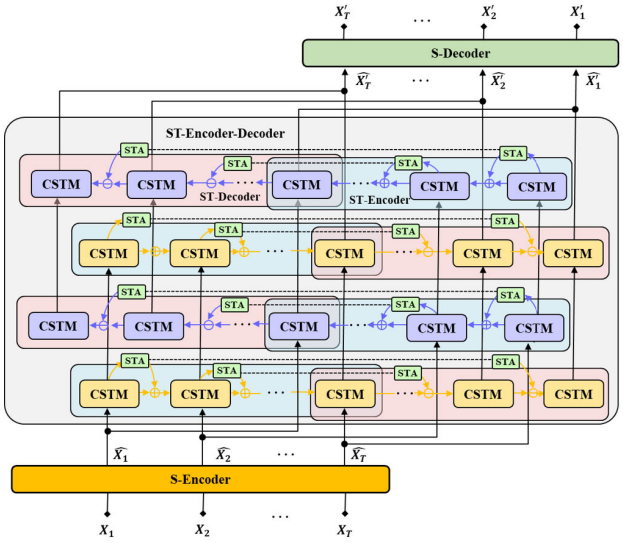

FIGURE 4. The bi-CRRN-ED architecture using S-Encoder, S-Decoder, ST-Encoder and ST-Decoder. STA represents ST-Attention.

1) bi-CRRN-E

Fig. 3 shows bi-CRRN-E which is composed of a S-Encoder, a S-Decoder, and a ST-Encoder. The S-Encoder extracts the spatial feature, $\hat{X}_t \in \mathbb{R}^{N_c \times N_h \times N_w}$ from the original image, $X_t \in \mathbb{R}^{N_c^{in} \times N_h^{in} \times N_w^{in}}$, where N_c (N_c^{in}), N_h (N_h^{in}) and N_w (N_w^{in}) are the number of channels, height and width of $\hat{X}_t$ (X_t), respectively. Then, these spatial features are passed through ST-Encoder layers which generates CSTM hidden values. These values from the forward and backward layers are concatenated and passed through the upper layer. At the top ST-Encoder-Decoder layer, the hidden values are concatenated and passed through the S-Decoder to generate a predicted output. We denote the forward and backward direction hidden states of the ST-Encoder by $\vec{E}_t^l$ and $\overleftarrow{E}_t^l$ at l-th layer and time frame t. The hidden state of the ST-Encoder is formulated as follows:

$$\vec{E}_t^l = CSTM(W_{1\times1} * [\vec{E}_t^{l-1}; \overleftarrow{E}_t^{l-1}], \vec{E}_{t-1}^l),$$

$$\overleftarrow{\boldsymbol{E}}_t^l = CSTM(\boldsymbol{W}_{1\times1} * [\vec{\boldsymbol{E}}_t^{l-1}; \overleftarrow{\boldsymbol{E}}_t^{l-1}], \overleftarrow{\boldsymbol{E}}_{t+1}^l). \quad (2)$$

We calculate the final ST-Encoder output by using the top ST-Encoder layer outputs. The output is formulated as follows:

$$\hat{\boldsymbol{X}}_t' = W_{1\times1}[\vec{\boldsymbol{E}}_t^L; \overleftarrow{\boldsymbol{E}}_t^L], \quad (3)$$

where $W_{1\times1} \in \mathbb{R}^{N_c\times 2N_c}$ represents one by one convolutional operation weight matrix. With the spatial feature $\hat{\boldsymbol{X}}_t' \in \mathbb{R}^{N_c^{out}\times N_h^{out}\times N_w^{out}}$, the S-Decoder generates the predicted output $\boldsymbol{X}_t' \in \mathbb{R}^{N_c\times N_h\times N_w}$.

The main advantage of bi-CRRN-E network is the reduced computation time because of the simpler network architecture. Therefore, it comes in handy to adopt in the industrial application. The absence of the ST-Decoder, however, may result in a low performance of the defect detection due to a lack of decoding temporal information. To cater for this issue, the bi-CRRN-E is designed as a fully connected CSTM architecture, in which each pair of forward and backward CSTM layers at time t is fully connected.

2) bi-CRRN-ED

Fig. 4 shows bi-CRRN-ED model, which is composed of S-Encoder, S-Decoder, ST-Encoder, and ST-Decoder. Unlike bi-CRRN-E, bi-CRRN-ED model is designed to use ST-Decoder for the improvement in the detection performance. After the input passes through the S-Encoder, the generated feature values are passed to the forward and backward direction CSTMs. The hidden states of ST-Encoder are formulated as follows:

$$\begin{aligned}\vec{\boldsymbol{E}}_t^l &= CSTM(\vec{\boldsymbol{E}}_t^{l-1}, \vec{\boldsymbol{E}}_{t-1}^l),\\ \overleftarrow{\boldsymbol{E}}_t^l &= CSTM(\overleftarrow{\boldsymbol{E}}_t^{l-1}, \overleftarrow{\boldsymbol{E}}_{t-1}^l).\end{aligned} \quad (4)$$

We denote the forward and backward direction hidden states of the ST-Decoder at l-th layer and time frame t by $\vec{\boldsymbol{D}}_t^l$ and $\overleftarrow{\boldsymbol{D}}_t^l$, respectively. The hidden states of ST-Decoder are respectively formulated as follows:

$$\begin{aligned}\vec{\boldsymbol{D}}_t^l &= CSTM(\vec{\boldsymbol{D}}_t^{l-1}, \vec{\boldsymbol{D}}_{t+1}^l),\\ \overleftarrow{\boldsymbol{D}}_t^l &= CSTM(\overleftarrow{\boldsymbol{D}}_t^{l-1}, \overleftarrow{\boldsymbol{D}}_{t+1}^l).\end{aligned} \quad (5)$$

Hidden states of the top ST-Decoder layer contain forward and backward sequential information. The predicted output is formulated as follows:

$$\hat{\boldsymbol{X}}_t' = W_{1\times1}[\vec{\boldsymbol{D}}_t^L; \overleftarrow{\boldsymbol{D}}_t^L], \quad (6)$$

where the top layer hidden states, $\vec{\boldsymbol{D}}_t^L$ and $\overleftarrow{\boldsymbol{D}}_t^L$ are concatenated in a channel-wise manner.

In addition, the spatio-temporal attention (ST-Attention) is used to further improve the long-term dependency. The hidden state of the ST-Encoder is expressed as E_t and the ST-Attention map is calculated as follows:

$$A_t = tanh(W_A * E_t), \quad (7)$$

where $W_A \in R^{1\times Nc\times Nk\times Nk}$ is the weight matrix of the convolutional operation and N_k denotes the kernel size of the convolution. A_t is replicated in a channel-wise manner to match the number of channels of the hidden state (E_t). The replicated ST-Attention map is added to the ST-Encoder and subtracted from the ST-Decoder. A_t acts as a shortcut path between encoder and decoder.

3) SUPERVISED bi-CRRN

The two designed bi-CRRNs are optimized to a supervised learning framework due to the high performance demands in industrial applications. In contrast, the traditional CRRN is a network developed for unsupervised anomaly detection. In the case of unsupervised learning, the network can be trained with a normal image dataset only. Therefore, it does not require dataset collected from anomalous welded targets. On the other hand, supervised learning models tend to be more accurate than unsupervised learning models. Thus, we design the bi-CRRN in a supervised learning framework with labeled binary images as ground truth. For the designed network, the loss value is calculated by comparing the output with the ground truth, where the pixel-wise binary cross entropy (BCE) loss function is used in the learning process. The whole process is summarized in Algorithm 1. Note that D_{tr} and D_{te} stand for training and validation datasets, respectively. Also $\mathcal{G}_\theta$ is a bi-CRRN model parameterized with θ.

Algorithm 1 Bi-CRRN Training and Validation Algorithm

Input: Image dataset D_{tr}, D_{te}
Output: An optimized bi-CRRN model $\mathcal{G}_{\theta^*}$ trained on D_{tr}

Phase 1 – Pre-processing phase

Images in the dataset D_{tr} and D_{te} are sliced with a T-sized window, since the network input should be consisted of T sequential images.

Phase 2 – Network training phase

Initialize θ
for Each epoch **do**
  for Each batch i **do**
    Calculate forward propagation $\hat{y}_{tr}^{(i)} = \mathcal{G}_\theta(x_{tr}^{(i)})$
    Calculate loss $\mathcal{L}$ by pixel-wise $BCE(\hat{y}_{tr}^{(i)}, y_{tr}^{(i)})$
    Update $\mathcal{G}_\theta$ with Adam optimizer using loss $\mathcal{L}$
  end for
end for

Phase 3 – Validation phase

for Each batch **do**
  Calculate forward propagation $\hat{y}_{te}^{(i)} = \mathcal{G}_{\theta^*}(x_{te}^{(i)})$
  Calculate accuracy and F1 score
end for

IV. EXPERIMENTS

A. EXPERIMENTAL SETUP

1) HARDWARE SETUP

Two types of equipment were designed and manufactured by DSEC, which is a marine engineering company in Korea.

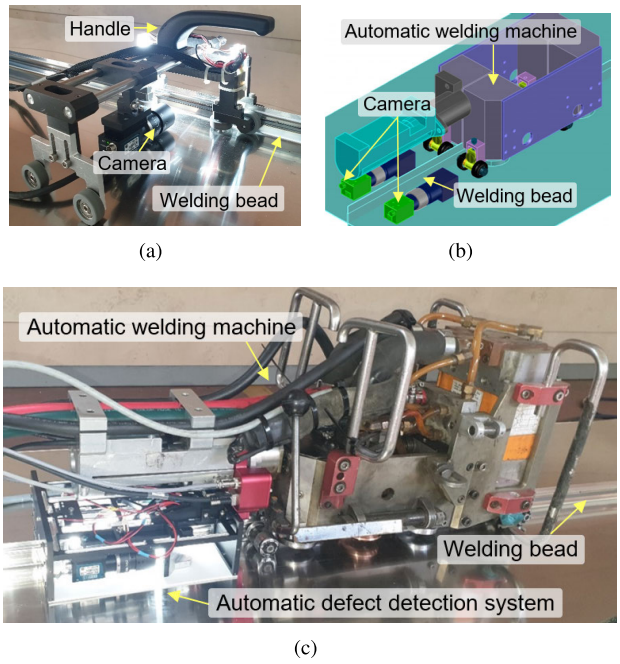

FIGURE 5. **Automatic welding defect detection hardware platforms. (a) Manual data acquisition equipment with a handle. (b) 3D drawing of the automatic welding defect detection equipment. (c) Welding machine with an automatic welding defect detection equipment.**

A manual training dataset acquisition equipment was manufactured as shown in Fig. 5(a). This equipment consists of camera, LED light, rollers moving along with the welding bead, and handle mounted on the top side. The operator can push the handle to collect the welding images. Based on the 3D drawing (Fig. 5(b)), automatic defect detection equipment (Fig. 5(c)) was manufactured, which is attached on an automatic welding machine. The equipment is composed of the same components as the manual acquisition equipment. Since the welding bead that the proposed equipment should monitor is at least 10 meters long, the defect detection equipment is attached to the automatic welding machine rather than fixed in a specific place. It means the welding process and the monitoring of the welding bead are carried out simultaneously.

2) DATASETS

We captured video clips of the welding bead because of the harsh experimental environment. The length of the welding bead is at least 10 meters long and the height of the bead is not fixed. Also, the experimental environment has vibration due to the movement of the automatic welding machine. In this environment, it is difficult to apply a single image-based defect detection network with static images. Therefore, the video input-based defect detection network was applied to monitor the welding defects.

The training dataset was obtained by the camera installed on the manual data acquisition equipment. In contrast, the validation dataset was obtained using the camera installed on the automatic defect detection equipment. Both cameras capture 20 frames per second, with each frame having $1,280 \times 720$ resolution. Considering the learning speed and storage capacity of our learning system, the image size was reduced to 160×90 resolution.

We generated the ground-truth images by labeling the actual locations of defects at pixel-level, with the guidance of a welding expert in the field. The annotated images of the dataset are shown in Fig. 6 (bottom row) where the green pixels indicate the defective region. We can see the various shapes of defect areas. If there is no defect, the annotated image is the same as the input image. The entire dataset consists of 310 video clips, which is a total of 75,420 frames. The dataset was divided into training and validation datasets with a ratio of 80% and 20%.

3) IMPLEMENTATION DETAILS

In the network architecture, the S-Encoder and S-Decoder are composed of two convolution layers, batch normalization, and ReLU layers. ST-Encoder and ST-Decoder consist of two CSTM layers. The kernel size was set to 5×5 filter. The sequential image frames were obtained by slicing image sequences with a window of size 10 and thus the number of the sequential image frames, T, was 10. Input and output were set to the two channels, and the rest layers were set to 64 channels.

For the network training, the number of epochs, batch size, and the learning rate are set to 150, 20, 0.0001, respectively. In addition, Adam [29] was used as an optimizer for the back propagation to learn the network parameters. The specifications of the workstation are Intel core i9-9900K, 32GB RAM, and the GPU is 4 GTX 1080Ti.

B. EVALUATION METRICS

To test our proposed network, we used accuracy, precision, recall, and F_β score as the evaluation metrics. *tp* (true positive) is the number that the network correctly detects the actual defect, *fp* (false positive) is the number that misclassifies the normal as a defect, *tn* (true negative) is the number that correctly detects the normal as normal, and finally *fn* (false negative) is the number misclassifying the defect as normal. From these counts, precision, recall, and F_β score are defined as follows:

$$\begin{aligned} \text{Precision} &= \frac{tp}{tp+fp} \\ \text{Recall} &= \frac{tp}{tp+fn} \\ F_\beta \text{ score} &= \frac{(1+\beta^2)\cdot \text{Precision} \cdot \text{Recall}}{(\beta^2 \cdot \text{Precision}) + \text{Recall}}, \end{aligned} \tag{8}$$

where the parameter β determines the weight of recall in the score.

C. PIXEL-LEVEL PERFORMANCE EVALUATION

To test the performance of the setup, an experiment for the real time defect detection at the pixel-level was carried out. We verified that the proposed bi-CRRN could successfully detect and localize the welding defects. The performance of

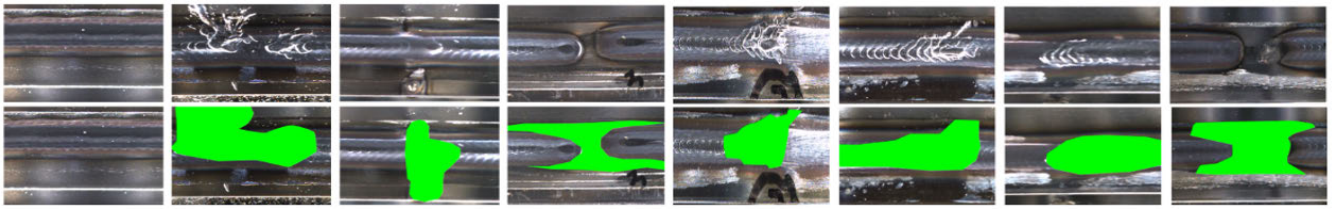

FIGURE 6. Examples of defect and non-defect image frames. The middle part of each frame with a comb pattern is the welding spot. The first column shows the non-defect frames, and the remaining columns denote frames containing defects. The first row is the original image and the second row denotes the labeled version of the first row.

the proposed bi-CRRN was compared with recent defect detection algorithms such as mask-RCNN [30], U-Net [31], DeepLabv3 [32], 3D-CNN [33], ConvLSTM, and CRRN. The network architectures of 3D-CNN and ConvLSTM, which exploit spatial and temporal information, were implemented based on the CRRN architecture. In the case of ConvLSTM network, the CSTMs of the ST-Encoder-Decoder were substituted for ConvLSTMs. On the other hand, instead of the ST-Encoder-Decoder, the autoencoder architecture was implemented for the 3D-CNN.

TABLE 1. Pixel-level welding defect detection performance.

Model	Accuracy (%)	Recall (%)	Precision (%)	F_1 score
mask-RCNN	96.93	52.15	48.26	0.5013
U-Net	97.61	65.97	57.81	0.6162
DeepLabv3	97.82	63.37	62.34	0.6285
3D-CNN	97.39	62.44	54.67	0.5830
Conv LSTM	97.86	70.69	64.44	0.6742
CRRN	98.16	73.64	64.38	0.6856
CRRN w/attn	98.14	70.71	67.29	0.6896
bi-CRRN-E	98.09	**80.30**	61.35	0.6956
bi-CRRN-ED	98.20	79.38	63.45	0.7095
bi-CRRN-ED w/attn	**98.31**	77.68	**66.04**	**0.7139**

We denote CRRN and bi-CRRN-ED both with the ST-Attention mechanism as CRRN w/attn and bi-CRRN-ED w/attn, respectively. Table 1 reports that mask-RCNN, U-Net, and DeepLabv3, which consider only spatial information without recurrent connection, present a low F_1 score. Since the dataset has a temporal property, ConvLSTM, CRRN, and bi-CRRN, which process spatial and temporal information, show better performance except for 3D-CNN. The bi-CRRN-ED, which handles the correlations across all the temporal meaning, presents the best F_1 score. Even though bi-CRRN-E is designed without the ST-Decoder, it has better performance than CRRN mainly due to the bidirectional memory connection and the fully connected memory cell at time t. Also, models with the ST-attention mechanism, which strengthens the long-term dependency, provide better detection performance than those without ST-Attention.

Additionally, we compared the precision-recall curves of the proposed bi-CRRN and other networks by sweeping over decision thresholds. The network output pixels are determined as defective pixels when the output values are greater than the decision threshold. As shown in Fig. 7, the bi-CRRN-ED w/attn outperforms other deep learning models.

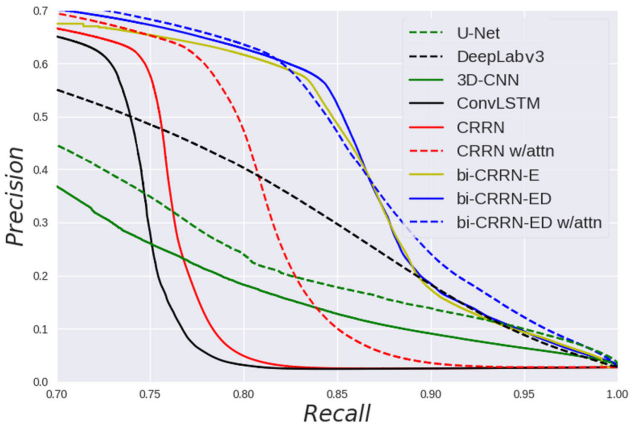

FIGURE 7. Precision-recall curve for defect detection (pixel-level).

TABLE 2. F_β scores for pixel-level welding defect detection.

Model	F_1 score	F_2 score	F_5 score	F_{10} score
mask-RCNN	0.5013	0.5132	0.5199	0.5211
U-Net	0.6162	0.6416	0.6561	0.7525
DeepLabv3	0.6285	0.6758	0.7308	0.7514
3D-CNN	0.5830	0.6287	0.7339	0.7852
Conv LSTM	0.6742	0.6755	0.6996	0.7385
CRRN	0.6856	0.7273	0.7657	0.7730
CRRN w/attn	0.6896	0.7389	0.7768	0.7839
bi-CRRN-E	0.6956	0.7641	0.8205	0.8316
bi-CRRN-ED	0.7095	0.7688	0.8296	0.8431
bi-CRRN-ED w/attn	**0.7139**	**0.7810**	**0.8612**	**0.8983**

The qualitative comparison is summarized in Fig. 8. It shows the input images with respect to the time axis, and the defect detection at the pixel-level. Pixels determined to be defective are shown in white, whereas the rest are in black. The results of 3D-CNN, ConvLSTM, and CRRN have some false positives in the non-defective pixels. In contrast, the proposed bi-CRRN-E and bi-CRRN-ED present more accurate defect detection results.

Meanwhile, we investigated imbalance ratios at the pixel-level to address the issues related to the imbalanced dataset [34]. Random masks were generated to adjust the imbalance ratio. Normal pixels were masked to verify how this imbalance affects the defect detection performance. We compared the performance of bi-CRRN-ED w/attn by sweeping the imbalance ratios (3:1 to 30:1). As shown in Table 3, since the imbalance ratio of the validation dataset

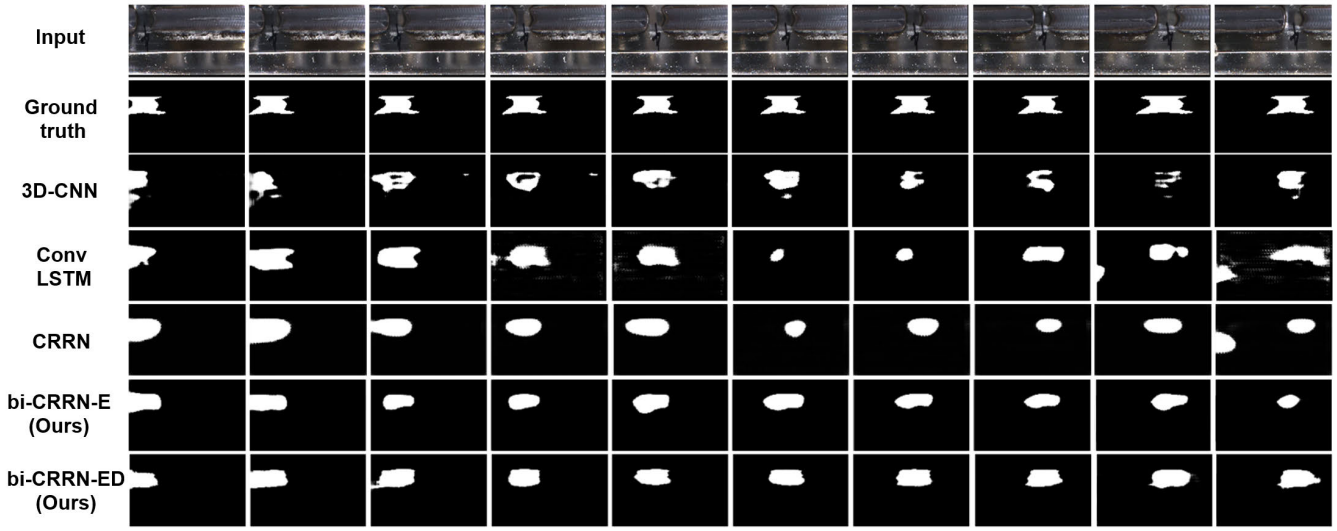

FIGURE 8. Qualitative comparison between 3D-CNN, ConvLSTM, CRRN, bi-CRRN-E and bi-CRRN-ED, showing pixel-level welding defect detection results.

TABLE 3. Pixel-level welding defect detection results with different imbalance ratios.

Model	Accuracy (%)	Recall (%)	Precision (%)	F_1 score
bi-CRRN-ED w/attn-3:1	97.52	**83.38**	52.79	0.6465
bi-CRRN-ED w/attn-10:1	97.90	76.44	58.73	0.6643
bi-CRRN-ED w/attn-15:1	98.16	77.13	63.34	0.6956
bi-CRRN-ED w/attn-30:1	**98.31**	77.68	**66.04**	**0.7139**

is about 30:1, the proposed network reports the best performance with the imbalance ratio of 30:1.

In the industrial field, recall is often more important than precision because the missed fault can lead to significant losses. Table 2 shows F_β scores for the pixel-level welding defect detection. Since F_β scores of bi-CRRN are higher than those of other models, it indicates that the proposed model attains a higher recall value.

D. FRAME-LEVEL PERFORMANCE EVALUATION

The frame-level defect detection was also performed based on the results of the bi-CRRN pixel-level defect detection. The operator can recognize immediately if the welding bead is defective through frame-level defect detection. The frame-level defect detection was calculated from the sequential image frames, where a T-sized window was employed. The image group is classified to be defective if the sum of all the defective pixels from the pixel-level output is larger than a threshold value as follows:

$$\sum_{i=1}^{T}\sum_{j=1}^{n} p_{ij} > \theta_{thres}, \tag{9}$$

where n is the number of pixels in one image frame, p is the pixel-level binary detection value, and θ_{thres} controls the sensitivity of the defect detection decision making. In this experiment, n and θ_{thres} were set to 14,400 and 1,000, respectively. Note that the number of the sequential image frames, T was set to 10.

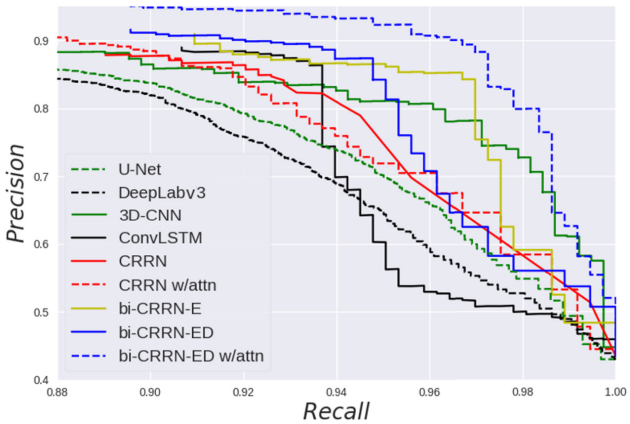

FIGURE 9. Precision-recall curve for defect detection (frame-level).

TABLE 4. Frame-level welding defect detection performance and computation time.

Model	Accuracy (%)	Recall (%)	Precision (%)	F_1 score	Computation time (s)
mask-RCNN	65.35	91.73	54.03	0.6801	0.191
U-Net	88.98	85.73	88.28	0.8699	0.077
DeepLabv3	89.05	84.57	89.37	0.8690	0.048
3D-CNN	89.20	91.91	84.62	0.8811	**0.021**
ConvLSTM	89.99	**92.68**	85.54	0.8897	0.094
CRRN	90.36	91.78	86.79	0.8921	0.106
CRRN w/attn	90.07	89.59	89.11	0.8934	0.118
bi-CRRN-E	91.67	91.51	89.54	0.9051	0.046
bi-CRRN-ED	92.02	90.96	90.71	0.9084	0.169
bi-CRRN-ED w/attn	**93.81**	90.68	**94.84**	**0.9271**	0.188

Fig. 9 shows the performance of the defect detection at frame-level. It shows that the frame-level defect detection using the proposed bi-CRRN-ED w/attn provides the best accuracy performance compared to other networks, as reported in Table 4. Similar to the pixel-level experimental result, the models with the ST-Attention mechanism show better performance than those without ST-Attention.

E. COMPUTATION TIME

Table 4 also shows the computation time for individual networks. By reducing the number of parameters in CRRN, CRRN and CRRN w/attn take similar computation time to the convLSTM. On the other hand, since our proposed bi-CRRN-E network is designed without ST-Decoder, the simplicity of the network architecture helps to further reduce the computation time. Thus, this network can be used in the fields that prioritize fast computation time over high accuracy. Although bi-CRRN-ED and bi-CRRN-ED w/attn have more computation time than other algorithms, these defect detection networks can also be used in the industrial site due to the outstanding performance.

V. CONCLUSION

In this paper, we proposed a novel deep learning network, bi-CRRN for spatio-temporal defect detection. We focused on two industrial demands: high detection accuracy and lightweight for less computation time. Thus, we designed two kinds of bi-CRRN architecture. Firstly, bi-CRRN-E network was designed to reduce the computation time. To maintain the defect detection performance, each memory cell is fully connected considering both forward and backward time sequences. Another network, bi-CRRN-ED was designed to get the high prediction performance. The efficiency of the designed networks was tested on the custom dataset collected from the hardware equipment developed exclusively for this purpose. The experimental results confirmed that both the bi-CRRN-E and the bi-CRRN-ED demonstrated higher accuracy on the pixel level as well as the frame level. Also, the computation time was verified for practical applications in the industrial fields through the experiments. The proposed network can be applied to other defect detection environments with video as input. When the network is applied to a building or plant surface crack detection, which is difficult to collect static images, higher performance can be expected than the single image-based defect detection models.

REFERENCES

[1] Z. He and Q. Liu, "Deep regression neural network for industrial surface defect detection," *IEEE Access*, vol. 8, pp. 35583–35591, 2020.
[2] H.-I. Lin and F. S. Wibowo, "Image data assessment approach for deep learning-based metal surface defect-detection systems," *IEEE Access*, vol. 9, pp. 47621–47638, 2021.
[3] C. Phua and L. B. Theng, "Semiconductor wafer surface: Automatic defect classification with deep CNN," in *Proc. IEEE REGION Conf. (TENCON)*, Nov. 2020, pp. 714–719.
[4] R. Augustauskas and A. Lipnickas, "Improved pixel-level pavement-defect segmentation using a deep autoencoder," *Sensors*, vol. 20, no. 9, p. 2557, Apr. 2020.
[5] Y. Huang, J. Jing, and Z. Wang, "Fabric defect segmentation method based on deep learning," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–15, 2021.
[6] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 802–810.
[7] Y. Wang, M. Long, J. Wang, Z. Gao, and S. Y. Philip, "PredRNN: Recurrent neural networks for predictive learning using spatiotemporal LSTMs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 879–888.
[8] Y.-H. Yoo, U.-H. Kim, and J.-H. Kim, "Convolutional recurrent reconstructive network for spatiotemporal anomaly detection in solder paste inspection," *IEEE Trans. Cybern.*, early access, Nov. 24, 2020, doi: 10.1109/TCYB.2020.3033798.
[9] Y. Liu, H. Zheng, X. Feng, and Z. Chen, "Short-term traffic flow prediction with Conv-LSTM," in *Proc. 9th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2017, pp. 1–6.
[10] C. Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu, W. Cheng, J. Ni, B. Zong, H. Chen, and N. V. Chawla, "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 1409–1416.
[11] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," 2015, *arXiv:1508.01991*. [Online]. Available: http://arxiv.org/abs/1508.01991
[12] E. N. Malamas, E. G. M. Petrakis, M. Zervakis, L. Petit, and J.-D. Legat, "A survey on industrial vision systems, applications and tools," *Image Vis. Comput.*, vol. 21, no. 2, pp. 171–188, Feb. 2003.
[13] A. Mital, M. Govindaraju, and B. Subramani, "A comparison between manual and hybrid methods in parts inspection," *Integr. Manuf. Syst.*, vol. 9, no. 6, pp. 344–349, Dec. 1998.
[14] Z. Hocenski, S. Vasilic, and V. Hocenski, "Improved Canny edge detector in ceramic tiles defect detection," in *Proc. 32nd Annu. Conf. IEEE Ind. Electron.*, Nov. 2006, pp. 3328–3331.
[15] A. Conci and C. B. Proença, "A computer vision approach for textile inspection," *Textile Res. J.*, vol. 70, no. 4, pp. 347–350, Apr. 2000.
[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[17] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.
[18] J. Tian, C. Morillo, M. H. Azarian, and M. Pecht, "Motor bearing fault detection using spectral kurtosis-based feature extraction coupled with K-nearest neighbor distance analysis," *IEEE Trans. Ind. Electron.*, vol. 63, no. 3, pp. 1793–1803, Mar. 2015.
[19] Z. Xue-Wu, D. Yan-Qiong, L. Yan-Yun, S. Ai-Ye, and L. Rui-Yu, "A vision inspection system for the surface defects of strongly reflected metal based on multi-class SVM," *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5930–5939, May 2011.
[20] J. Mirapeix, P. B. García-Allende, A. Cobo, O. M. Conde, and J. M. López-Higuera, "Real-time arc-welding defect detection and classification with principal component analysis and artificial neural networks," *NDT E Int.*, vol. 40, no. 4, pp. 315–323, 2007.
[21] S. Faghih-Roohi, S. Hajizadeh, A. Núñez, R. Babuska, and B. D. Schutter, "Deep convolutional neural networks for detection of rail surface defects," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2016, pp. 2584–2589.
[22] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 5, pp. 361–378, May 2017.
[23] C. Hu and Y. Wang, "An efficient convolutional neural network model based on object-level attention mechanism for casting defect detection on radiography images," *IEEE Trans. Ind. Electron.*, vol. 67, no. 12, pp. 10922–10930, Dec. 2020.
[24] G. Kang, S. Gao, L. Yu, and D. Zhang, "Deep architecture for high-speed railway insulator surface defect detection: Denoising autoencoder with multitask learning," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 8, pp. 2679–2690, Aug. 2018.
[25] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2016.
[26] J. Lee and K. Um, "A comparison in a back-bead prediction of gas metal arc welding using multiple regression analysis and artificial neural network," *Opt. Lasers Eng.*, vol. 34, no. 3, pp. 149–158, Sep. 2000.
[27] Y. Feng, Z. Chen, D. Wang, J. Chen, and Z. Feng, "DeepWelding: A deep learning enhanced approach to GTAW using multisource sensing images," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 465–474, Jan. 2019.
[28] P. Sassi, P. Tripicchio, and C. A. Avizzano, "A smart monitoring system for automatic welding defect detection," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9641–9650, Dec. 2019.
[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[30] L. Attard, C. J. Debono, G. Valentino, M. D. Castro, A. Masi, and L. Scibile, "Automatic crack detection using mask R-CNN," in *Proc. 11th Int. Symp. Image Signal Process. Anal. (ISPA)*, Sep. 2019, pp. 152–157.
[31] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.
[32] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: http://arxiv.org/abs/1706.05587
[33] L. Zhang, G. Zhu, P. Shen, J. Song, S. A. Shah, and M. Bennamoun, "Learning spatiotemporal features using 3DCNN and convolutional LSTM for gesture recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 3120–3128.
[34] P. Tripicchio, G. Camacho-Gonzalez, and S. D'Avella, "Welding defect detection: Coping with artifacts in the production line," *Int. J. Adv. Manuf. Technol.*, vol. 111, nos. 5–6, pp. 1659–1669, Nov. 2020.

YOUNG-MIN KIM received the B.S. degree in mechanical and electronic control engineering from Handong University, Pohang, Republic of Korea, in 2013, and the M.S. degree in robotics program from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea, in 2015, where he is currently pursuing the Ph.D. degree. His current research interests include anomaly detection and optimization methods for industrial fields.

IN-UG YOON received the B.S. and M.S. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea, in 2016 and 2018, respectively, where he is currently pursuing the Ph.D. degree. His current research interests include anomaly detection, learning algorithms, and computational memory systems.

HYUN MYUNG (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea, in 1992, 1994, and 1998, respectively, all in electrical engineering. He was a Senior Researcher with the Electronics and Telecommunications Research Institute, Daejeon, from 1998 to 2002, the CTO and the Director with the Digital Contents Research Laboratory, Emersys Corporation, Daejeon, from 2002 to 2003, and a Principle Researcher with Samsung Advanced Institute of Technology, Yongin, Republic of Korea, from 2003 to 2008. From 2008 to 2018, he was a Professor with the Department of Civil and Environmental Engineering, KAIST, where he is currently a Professor with the School of Electrical Engineering, KI-Robotics, KI-AI, and the Head of the KAIST Robotics Program. His current research interests include structural health monitoring using robotics, artificial intelligence, simultaneous localization and mapping, robot navigation, machine learning, deep learning, and swarm robots.

JONG-HWAN KIM (Fellow, IEEE) received the Ph.D. degree in electronics engineering from Seoul National University, Republic of Korea, in 1987. Since 1988, he has been with the School of Electrical Engineering, KAIST, Republic of Korea, where he is leading the Robot Intelligence Technology Laboratory as KT Endowed Chair Professor. He is the Director for both of KoYoung-KAIST AI Joint Research Center and the Machine Intelligence and Robotics Multi-Sponsored Research and Education Platform. He has authored five books, ten edited books, and around 450 refereed articles in technical journals and conference proceedings. His research interests include intelligence technology, machine intelligence learning, and AI robots.