This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388

A Fine-Grained Attention Model for High Accuracy Operational Robot Guidance

Yinghao Chu, Daquan Feng, Zuozhu Liu, Lei Zhang, Zizhou Zhao, Zhenzhong Wang, Zhiyong Feng, and Xiang-Gen Xia

***Abstract*—Deep learning enhanced Internet of Things (IoT) is advancing the transformation towards smart manufacturing. Intelligent robot guidance is one of the most potential deep learning+IoT applications in the manufacturing industry. However, low costs, efficient computing, and extremely high localization accuracy are mandatory requirements for vision robot guidance, particularly in operational factories. Therefore in this work, a low-cost edge computing based IoT system is developed based on an innovative Fine-Grained Attention Model (FGAM). FGAM integrates a deep-learning based attention model to detect the Region Of Interest (ROI) and an optimized conventional computer vision model to perform fine-grained localization concentrating on the ROI. Trained with only 100 images collected from real production line, the proposed FGAM has shown superior performance over multiple benchmark models when validated using operational data. Eventually, the FGAM based edge computing system has been deployed on a welding robot in a real-world factory for mass production. After the assembly of about 6000 products, the deployed system has achieved averaged overall process and transmission time down to 200 ms and overall localization accuracy up to 99.998%.**

***Index Terms*—Internet of Things, Edge Computing, Deep Learning, Attention Mechanism, Fine-grained Image Analysis, Smart Manufacturing, Robot Guidance**

I. Introduction

Deep learning enhanced Internet of Things (IoT) [1], [2] is envisioned in order to play a vital role in the domain of smart manufacturing [3]–[6] to achieve substantial advancements in terms of quality control, productivity, and efficiency [7], [8]. Representative deep learning enhanced IoT applications for smart manufacturing include, but not limited to, automatic fault diagnosis, defect prognosis, surface quality inspection, and robot guidance [9]–[15]. Among these applications, smart vision guidance system is capable of improving the performance of identification accuracy, localization accuracy, obstacle avoidance, and human-robot collaboration for manufacturing robots [15]–[18]. Development and deployment of the vision guidance systems are usually based on edge computing enabled IoT systems [19]–[21]. Therefore, the deployed systems are able to operate in isolated mode without internet access and guarantee the production safety and data security for factories [22]–[24].

However, there are several remaining challenges to develop vision guidance method on edge computing based IoT system for manufacturing users. First, factories demand low-cost solutions to maintain their market competitiveness. Small-scale machinery and manufacturing factories are particularly sensitive to costs [25]. Second, the embedded algorithm of an operational system must be highly computational efficient [26]. The efficiency of the edge computing directly affects the final hourly production rate and the revenue of the factories. Third, a very high-level accuracy of vision localization is mandatory for guided manufacturing operations [15]. Guided robot movements using incorrect object positions may not only increase the defective product rate, but also are potentially dangerous to the system hardware and human operators. Consequently, most manufacturing industries have strict requirements on the accuracy of vision localization. For example, the localization deviation from the true object for spot welding should be less than 1 millimeter in automotive industry [27]. However, available vision localization methods may not completely suit the complex environments in factories. For example, conventional computer vision methods are sensitive to variations in illumination and background noise [28]. Similarly, the localization precision of deep learning detection may degenerate due to jittering effects [29], [30]. More details about the state-of-the-art vision localization methods will be discussed in Section II.

Therefore in this work, a Fine-Grained Attention Model (FGAM) is developed based on edge computing based IoT to enable accurate, efficient, and low-cost 2D robot guidance. FGAM first uses a deep learning based attention model for Region Of Interest (ROI) detections and then uses an optimized conventional computer vision method for fine-grained pixel-

This work was supported in part by the National Science and Technology Major Project under Grant 2020YFB1807601, the Shenzhen Science, and Technology Program under Grants JCYJ20210324095209025, and the Open Foundation of State key Laboratory of Networking and Switching Technology (Beijing University of Posts and Telecommunications) under Grant SKLNST-2020-1-11.

Y. Chu is with Department of Advanced Design and Systems Engineering, City University of Hong Kong, Kowloon, Hong Kong

D. Feng is with the Shenzhen Key Laboratory of Digital Creative Technology, the Guangdong Province Engineering Laboratory for Digital Creative Technology, College of Electronics and Information Engineering, Shenzhen University, Shenzhen 518060, China. Daquan Feng is also with State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China. D. Feng is the corresponding author (e-mail: fdquan@gmail.com).

Z. Liu is with ZJU-UIUC Institute, Zhejiang University, Hangzhou, Zhejiang, 31002, China.

L. Zhang is with James Watt School of Engineering, University of Glasgow, Glasgow, G12 8QQ, UK.

Z. Zhao is with AIATOR Co., Ltd., Block 5, Room 222, Qianwanyilu, Qianhai, Shenzhen, 518060, China.

Z. Wang is with the Technical Management Center, China Media Group, Beijing 100020, China.

Z. Feng is with the Key Laboratory of the Universal Wireless Communications, Beijing University of Posts and Telecommunications, Beijing, 100020, China.

X.-G. Xia is with the Department of Electrical and Computer Engineering, University of Delaware, Newark, DE 19716, USA.

level localizations. The attention model uses modified Tiny-YOLOv3 backbone to optimize the computing efficiency. With the predicted ROI from the attention model, a fine-grained localization model is developed to consistently predict accurate locations of the target object. Fine-grained methods are mostly proposed to distinguish subordinate-level categories [31]–[33]. To the best knowledge of the authors, few researches have investigated the fine-grained methods for localization tasks, particularly for edge computing IoT system.

The major contributions of this work are summarized as follows.

1) Instead of a proof-of-concept demo in laboratory environment, a low-cost edge computing based IoT system is developed for robotic vision guidance in a real-world welding factory. On one hand, the automatic welding solution can greatly reduce chance of health hazards due to the tough working condition. On the other hand, it can help to alleviate the pressure of the skilled labor shortage that would adversely affect the production and revenue. The developed system is capable of operating in a noisy, complex, and hazardous environment but has shown excellent and robust performance during the real-time mass production. The deployed system can theoretically assemble more than 1000 products each day, and this productivity is equivalent to that of 2 full-time skilled human workers.
2) An innovative fine-grained attention model is developed for the IoT edge server to achieve highly accurate and robust vision localization. This model integrates a deep-learning object detection method for ROI detection and an optimized conventional computer vision method for fine-grained localization. The proposed model is capable of operating in complex factory environments. Trained using a small labelled dataset of only 100 images, this proposed model has shown a localization accuracy up to 99.998% in nearly 96000 vision guided welding operations during the real-time mass production. This proposed model is particularly useful for the manufacturing scenarios, in which data labelling is costly but a high level of localization accuracy is mandatory.
3) A new object detection backbone is proposed for ROI detection to maximize the computational efficiency. Comparing to the original Tiny-YOLOv3 backbone, the process and response speed of the new backbone improve from about 20 FPS to 52 FPS (for 1024×1280 images) with the minimum compromise in the detection accuracy.

The rest of the paper is organized as follows: the related work is presented in Section II; the system model and problem formulation are presented in Section III; the details to develop the FGAM are presented in Section IV; the experiments and results are presented and discussed in Section V and the conclusions are presented in Section VI.

II. Related Work

To achieve both accurate and robust vision localization for robot guidance, different methods have been proposed and discussed in the literature. For example, 3-D vision systems, such as structured light sensors [34], laser tracker [35], stereo cameras [36], and ToF camera [18], have been used in many robot guidance applications. However, 3-D vision systems are relatively expensive but small-scale factories (e.g. the investigated scenario in this work) are particularly sensitive to costs. Besides, the lifetime of 3-D systems will be noticeably shortened in harsh factory environments, which will further increase the maintenance and replacement costs. In addition, 3-D based systems raise another concern over the production efficiency. Edge computing servers usually have limited resource for sophisticated guidance algorithm. 3-D vision systems may take over a second to generate the point cloud or depth map, which will significantly decrease the hourly production rate and the overall revenue of the factories.

Therefore, low-cost 2-D vision systems are competitive solutions for scenarios with flat surfaces. 2-D vision systems could capture and process an image in milliseconds. Traditional approaches to perform localization in 2-D images include, but not limited to, template matching [37], shape fitting [38], edge-based matching [39], Scale-Invariant Feature Transform (SIFT) [40], Speeded Up Robust Features (SURF) [41], Features from Accelerated Segment Test (FAST) [42]. These traditional methods are well-established, transparent, and optimized for computational efficiency [43]. However, the accuracy and robustness of these methods are still improvable in the complex manufacturing environment.

In recent years, Convolutional Neural Network (CNN) dominates the interests of computer vision researches [44]–[46]. For classification and detection applications, CNN have shown superior performance over the above traditional methods in terms of accuracy, generalization, and robustness [47]–[50]. To find the locations of target objects in input images, object localization methods [51] have been proposed based on CNNs. These object localization methods first implement CNNs to extract image features and then apply boundary regressors to derive object positions using the extracted image features [52]. However, object localization methods predict fixed number of objects during the inference phase, and the number of predictions is determined and fixed during the training phase. To address this shortage, object detection methods are proposed to localize and classify all objects in an image. There are two major categories of object detection methods: region proposal based methods and regression/classification based methods [29]. Examples of region proposal based methods include, but not limited to, R-CNN [53], SPP [54], Fast R-CNN [55], Faster R-CNN [56], R-FCN [57], FPN [58], and examples of regression/classification based methods include, but not limited to, MultiBox [59], SSD [60], YOLO [61], DSSD [62].

As one of the representative regression-based methods, You Only Look Once (YOLO) method [61] predicts bounding boxes and class probabilities of target objects directly from an input image in one evaluation. As a result, YOLO achieves extremely fast speed up to 45 Frames Per Second (FPS) [61] when evaluated using the COCO dataset [63]. In addition, YOLO can be optimized end-to-end directly on detection performance, which is convenient for in-field deployments. In

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388

2018, YOLOv3 [64] is evolved from YOLO [61] and YOLOv2 [65]. YOLOv3 predicts bounding boxes on different scales using feature pyramid networks [66]. Therefore, YOLOv3 mitigates the loss of fine features during the downsampling of the input and is particularly useful to detect small objects. YOLOv3 is a potential solution to operational localization tasks because of its speed and ease of use. However, the bounding boxes predicted by YOLOv3 or other object detection methods may jitter around the ground-truths. The jittering effect is usually originated from inherent pixel noise of camera sensor or improper aggregation of the proposed bounding boxes [29], [30]. The bounding box jittering may result in localization deviations. Therefore, the jittering effect is a major challenge to apply object detection methods in high-precision machining and manufacturing scenarios.

To further enhance the consistence and robustness of deep learning models, attention mechanisms have been proposed. Attention models mimic the attention mechanisms of the human who focus on specific aspects of a complex input [67]. Therefore, a complicated problem is divided by attention models into smaller and simpler tasks that are processed sequentially [68]. Attention mechanisms have been widely employed in both natural language processing [68] and image analysis researches [69]–[72]. For image analysis, there are two main types of attention mechanisms: soft attention and hard attention [73]. For soft attention, a sophisticated group of filters is used to create a blurring effect that the ROI is in focus while the surrounding is faded or blurred. For hard attention, only the ROI is further analyzed and the entire background is discarded. In this work, the hard attention mechanism [73] is adopted to avoid the background noise (will be described in Section V-A). At the request of the investigated factory, the proposed attention model uses modified Tiny-YOLOv3 backbone to minimize the image analysis time.

III. System model and problem formulation

In this section, we first introduce the system model and then describe the optimization problem to maximize the localization accuracy for vision robot guidance.

A. System model

For a typical vision guided manufacturing robot, the applied edge computing system consists of 4 major components: an edge computing server, an imaging system, a user interface, and the executive manufacturing equipments (this work includes a manufacturing robot, a welder, and a rotary welding positioner). The schematic diagram of the proposed system is presented in Fig. 1. For operation, the edge computing server first initializes the system based on the start signal and the set parameters from the user interface. Then, the edge computing server pushes orders to other system components, directs operation of tasks, monitors and controls the process. The vision localization and robot guidance tasks are operated by the imaging system and the edge computing server. With the start signal, the edge computer server will direct the imaging system to capture the image x of the product to be processed. The captured image x is an $H \times W$ matrix, where H and W are the height and width of the image, respectively. The value of each matrix element $I_{h,w} \in [0, 255]$ represents the pixel intensity. Then, the imaging system forwards the captured images to the edge computing server. The proposed fine-grained attention model is embedded in the edge computing system, which analyzes x to generate the output $\hat{y}^j_{\text{h}} \in \{1, H\}$ and $\hat{y}^j_{\text{w}} \in \{1, W\}$. $\hat{y}^j_{\text{h}}$ and $\hat{y}^j_{\text{w}}$ are the coordinate predictions in the directions of height and width, respectively. $j \in \{1, 2, ..., k\}$, and k denotes the total number of points of interest (in this work is equal to 2) to be identified and localized in the image.

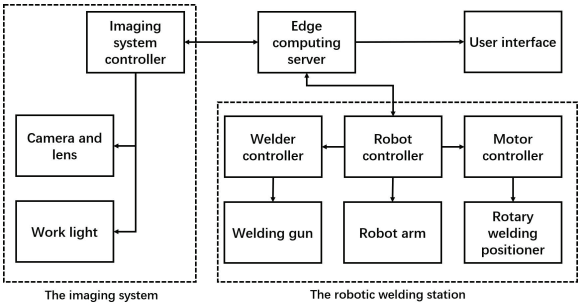


Fig. 1. The schematic diagram of the edge based IoT enhanced vision guided welding robot system.

B. Problem description

This work develops a model to perform high-accuracy vision localization based on a small dataset. In the following, x_i denotes an input image; $X = \{x_1, x_2, ..., x_n\}$ denotes the training set of input images; n denotes the number of images in the set X. For each x_i, there are j sets of labels $(y^j_{\text{h},i},\ y^j_{\text{w},i})$ that describe the positions of the points to be localized in x_i. $x_{\text{new}} \notin X$ denotes a new image. $(y^j_{\text{h,new}},\ y^j_{\text{w,new}})$ denotes the corresponding sets of labels for the new image. We consider an attention model $f(\cdot)$ to extract ROIs $\hat{v}^j_i$ from x_i, then $\hat{v}^j_i \triangleq f(x_i)$. Similarly, $\hat{v}^j_{\text{new}} \triangleq f(x_{\text{new}})$. $g(\cdot)$ denotes a fine-grained localization model that takes $\hat{v}^j_i$ as the input and generates $(\hat{y}^j_{\text{h,new}},\ \hat{y}^j_{\text{w,new}})$ as the final predictions. Then, $(\hat{y}^j_{\text{h},i}, \hat{y}^j_{\text{w},i}) \triangleq g(v^j_i)$. Similarly, $(\hat{y}^j_{\text{h,new}}, \hat{y}^j_{\text{w,new}}) \triangleq g(v^j_{\text{new}})$. Based on the given training set X, our objective is to develop $f(\cdot)$ and $g(\cdot)$ in order to maximize the localization accuracy by minimizing the difference between labels and predictions $(y^j_{\text{h,new}} - \hat{y}^j_{\text{h,new}})^2 + (y^j_{\text{w,new}} - \hat{y}^j_{\text{w,new}})^2$. In the following, we describe the proposed fine-grained attention model to solve these problems.

IV. Fine-grained attention model

To localize the points of interest with extremely high accuracy and speed, we propose a Fine-Grained Attention Model (FGAM). FGAM integrates a modified Tiny-YOLOv3 method [74] with an Improved Template Matching Model (ITMM) [37]. First, a modified Tiny-YOLOv3 is developed as the attention model $f(\cdot)$ to detect the attention mask or ROI patches v^j_i from the original images x_i. Then, the ITMM $g(\cdot)$ concentrates solely on the ROI patches v^j_i to perform fine-grained localization for the points of interests $(\hat{y}^j_{\text{h}},\ \hat{y}^j_{\text{w}})$. Details of the model development procedures are described in the followed subsections.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388

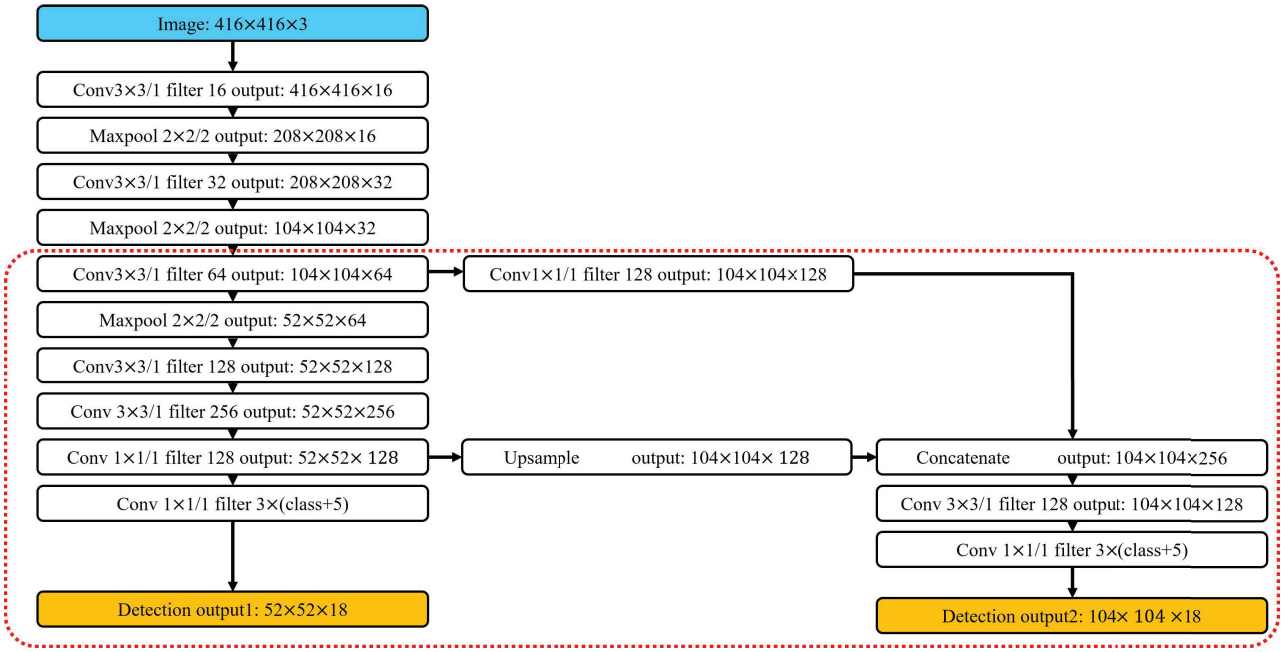


Fig. 2. The schematic diagram of the proposed attention model. This attention model is based on the modified backbone of Tiny-YOLOv3 and the modified layers are marked in the red dash box. The input image is marked by the blue box. The final output tensors from the attention model are marked by the orange boxes.

A. Attention model based on modified Tiny-YOLOv3

The attention model in this work is developed based on Tiny-YOLOv3, which is a simplified and fast version of YOLOv3. Tiny-YOLOv3 has shown both high detection accuracy and speed for both large and small objects [75], [76]. As discussed in Section V-A, the application scenario of this work demands that the overall vision localization process should be less than 200ms in real-time operation. However, the overall process contains a series of sub-processes, such as image capturing and transmission, image preprocessing, ROI localization, fine-grained localization, coordinate derivation and transformation. Therefore, only a small fraction of the 200ms is available to the attention model, which should generate the ROIs from each input image in 50ms based on preliminary calculations. Besides, the computational resource of the edge computing system is limited due to the consideration of costs. As a result, the Tiny-YOLOv3 [64], [77] is one of the few methods that could satisfy the efficiency demands of real-time operation.

The architecture of the proposed attention model is presented in Fig. 2. Similar to other YOLO methods, Tiny-YOLOv3 employs single neural network to directly predict bounding boxes with class probabilities and confidences in one evaluation [61]:

$$\hat{v}_i^j = (\hat{p}_\mathrm{c}, \hat{b}_\mathrm{h}, \hat{b}_\mathrm{w}, \hat{b}_\mathrm{H}, \hat{b}_\mathrm{W}, \boldsymbol{c})^j, \tag{1}$$

where j is the number of bounding boxes identified in the image i, $\hat{p}_\mathrm{c}$ is the probability that the bounding box contains an object, ($\hat{b}_\mathrm{w}$,$\hat{b}_\mathrm{h}$) is the center coordinate of the bounding box, $\hat{b}_\mathrm{H}$ and $\hat{b}_\mathrm{W}$ are height and width of the bounding box, respectively, $\boldsymbol{c}$ is the vector of object classes. Therefore, the number of elements n_v in $\hat{v}_i^j$ is equal to 5+number of classes in $\boldsymbol{c}$.

To perform detection, each input image is first divided into $S \times S$ grids. S is equal to the size of input image divided by a downsampling rate. For each grid cell, multiple bounding boxes can be generated based on bounding box priors, which are determined by k-means clustering [78] to improve the detection accuracy [64], [76]. Therefore, the size of the output detection tensor is

$$S \times S \times (B \times (n_v)), \tag{2}$$

where B is the number of bounding boxes priors.

To reduce the loss of fine features during the downsampling, feature pyramid network [66] is employed by the attention model to predict bounding boxes on different scales [64]. Different scales are given by downsampling the dimensions of the input image by the different downsampling rates. The number of scales for the attention model is 2, which equals to that of the Tiny-YOLOv3 [64]. Therefore, two detection tensors will be generated by the attention model as shown in the orange boxes in Fig. 2.

The Tiny-YOLOv3 backbone of the attention model has been optimized in order to further improve the computational efficiency and the detection accuracy. Implemented modifications are marked by the red box in Fig. 2 and are summarized as follows.

- The downsampling rate for the two detection scales are reduced from 32 and 16 to 8 and 4, respectively. As a result, the Tiny-YOLOv3 backbone of the attention model is further simplified, and the detection speed is enhanced from about 20 FPS to 52 FPS when evaluating on the 1024×1280 images from the vision system. The

reduction of downsampling rate will adversely affect the detection accuracy for large objects. Fortunately, the ROIs of welding joints in this work are relatively small objects in all captured images. Therefore, this new simplified architecture is employed by the attention model to further improve the computational efficiency.

- A few of layers with 1×1 convolution filters are added in front of the upsampling, concatenate, and final detection layers. The 1×1 layers add non-linearity and therefore enhance the learning of deeper or more abstract features without significant increasing of the computational cost [79]. Therefore, these new layers are used to improve the detection accuracy.

In this work, there is only 1 class that needs to be detected. Therefore, the number of classes is set to 1 to represent the ROI of the welding joints. Besides, the number B of bounding box priors is set to 3 as suggested in [64]. Therefore, the output detection tensors on two scales are $52 \times 52 \times 18$ and $104 \times 104 \times 18$, respectively. Eventually, bounding boxes are derived from these two output detection tensors for each input image. However, multiple bounding boxes might be predicted for the same target. Therefore, Non-Maximum Suppression (NMS) [80], [81] is employed to filter the boxes and output the optimal one.

The implementation of NMS employs Intersection over Union (IoU) as an important metric to evaluate the predicted bounding boxes (the ROIs) and to select the best bounding box [76]. IoU is defined as the overlap ratio between a predicted bounding box and the ground-truth/labelled bounding box:

$$\text{IoU} = \frac{S_{\text{o}}}{S_{\text{u}}}, \tag{3}$$

where S_{o} is the area of the intersection between predicted bounding box and the ground-truth bounding box, and S_{u} is the area of the union encompassed by both the predicted bounding box and the ground-truth bounding box. Therefore, high IoU indicates that the predicted bounding box well matches the ground-truth bounding box.

To filter multiple prediction boxes with both high confidence scores and IoUs, the process of NMS includes four steps described as follows.

1) With a list of proposal boxes l_{p}, remove the proposal $\hat{v}$ with the highest confidence score from l_{p} and append it to the list of filtered boxes l_{f}.
2) Compute the IoU of $\hat{v}$ with all proposals that are still in l_{p}. If the IoU of a proposal is higher than a predefined threshold θ, then discard this proposal from l_{p}.
3) Find the next proposal $\hat{v}$ with the highest confidence score in the remaining l_{p}, and repeat the above procedures until l_{p} is empty.
4) Output the l_{f}.

In practice, threshold θ is usually set to 0.5 for NMS purpose [61].

B. Training and optimization of the attention model

Back propagation training [82]–[84] is employed in this work to estimate the parameters of the attention model described above. The procedures for the back propagation training is presented as follows.

1) Initialize the trainable parameters of the deep learning network (weights w and bias b) randomly. Initialization methods, such as the Xavier method [85] or the HE method [86], can be employed to improve the robustness and efficiency of the training process.
2) Calculate the outputs $\hat{p} = f(w, b, X)$ based on the initialized parameters and the training data input X.
3) Calculate the final output errors ε using data labels and loss functions. The loss function for ROI detection should consider both localization and classification accuracies. Therefore, the YOLO Loss [61], [87] is used in this work, which composes of the classification loss E_{cls}, the coordinate loss E_{coord}, and the confidence loss E_{con}:
$$\varepsilon = E_{\text{cls}} + E_{\text{coord}} + E_{\text{con}}. \tag{4}$$
The classification loss is defined as:
$$E_{\text{cls}} = \sum_{i=1}^{S^2}\sum_{j=1}^{B}\mathbf{1}_{ij}^{obj}\sum_{c\in classes}(p_i(c) - \hat{p}_i(c))^2, \tag{5}$$
where S^2 is the number of grids in the output tensor, B is the number of bounding boxes in each grid, $\mathbf{1}_{ij}^{obj}$ is 1 if the object falls in the $i-th$ grid of the $j-th$ bounding box, otherwise $\mathbf{1}_{ij}^{obj}$=0, c is a target class to be detected, $p_i(c)$ is the true probability that the object of class c is in the grid i, and $\hat{p}_i(c)$ is the predicted value that the object of class c is in the grid i.
The coordinate loss is defined as:
$$\begin{aligned} E_{\text{coord}} = {} & \lambda_{\text{coord}}\sum_{i=1}^{S^2}\sum_{j=1}^{B}\mathbf{1}_{ij}^{obj}\left[(b_{\text{h}}^i - \hat{b}_{\text{h}}^i)^2 + (b_{\text{w}}^i - \hat{b}_{\text{w}}^i)^2\right] \\ & +\lambda_{\text{coord}}\sum_{i=1}^{S^2}\sum_{j=1}^{B}\mathbf{1}_{ij}^{obj}\left[(b_{\text{H}}^i - \hat{b}_{\text{H}}^i)^2 + (b_{\text{W}}^i - \hat{b}_{\text{W}}^i)^2\right], \end{aligned} \tag{6}$$
where λ_{coord} is the weight of the coordinate loss, $\hat{b}_{\text{h}}^i, \hat{b}_{\text{w}}^i, \hat{b}_{\text{H}}^i, \hat{b}_{\text{W}}^i$ are the values of center coordinate, height, width of the predicted bounding box, respectively, $b_{\text{h}}^i, b_{\text{w}}^i, b_{\text{H}}^i, b_{\text{W}}^i$ are the values of center coordinate, height, width of true bounding box, respectively.
The confidence loss is defined as:
$$\begin{aligned} E_{\text{con}} = {} & \sum_{i=1}^{S^2}\sum_{j=1}^{B}\mathbf{1}_{ij}^{obj}(C_i - \hat{C}_i)^2 \\ & +\lambda_{\text{noobj}}\sum_{i=1}^{S^2}\sum_{j=1}^{B}(1 - \mathbf{1}_{ij}^{obj})(C_i - \hat{C}_i)^2, \end{aligned} \tag{7}$$
where λ_{noobj} is a weight factor reducing the loss when the background is detected, C_i is the labelled confidence, and $\hat{C}_i$ is the predicted confidence.
4) Based on the YOLO loss ε^T at the training timestep T, calculate the gradients $\delta_w^T = \partial\varepsilon^T/\partial w$ and $\delta_b^T = \partial\varepsilon^T/\partial b$ for all neurons in the network, and update the weight parameters $w^{T+1} = w^T - \alpha\delta_w^T$ and the bias parameters

$b^{T+1} = b^T - \alpha\delta_b^T$, where α is the learning rate coefficient.

5) Repeat steps 2-4 recursively until the difference in error changes $\Delta\varepsilon = \varepsilon^T - \varepsilon^{T-1}$ less than a pre-defined tolerance parameter or the training reaches the maximum number of epochs. Afterward, output the weights and bias determined in the final round.

Mini Batch Gradient Descent (MBGD) method with a batch size of 32 is employed in this work to increase the training efficiency [88]. To enhance the efficiency and robustness of MBGD training, the ADaptive Moment Estimation (ADAM) algorithm [89] is used as the optimizer for training. In practice, ADAM is one of the most popular training optimizers [25] and is described as follows. First, the 1^{st} moment vector u_1 is calculated for trainable network parameters p (such as weights w and biases b) using an exponential decay rate β_1 that ranges from 0 to 1:

$$u_1^T = \beta_1 u_1^{T-1} + (1-\beta_1)\delta^T, \tag{8}$$

where $\delta = \partial\varepsilon/\partial p$ is the gradient calculated using the loss function for all neurons in the network, and T is the timestep. Then, the 2^{nd} moment vector u_2 is calculated using another exponential decay rates β_2 that ranges from 0 to 1:

$$u_2^T = \beta_2 u_2^{T-1} + (1-\beta_2)\delta^{T^2}. \tag{9}$$

Then, the bias-corrected first moment vector $\hat{u}_1$ is calculated

$$\hat{u}_1^T = u_1^T/(1-\beta_1^T). \tag{10}$$

Afterward, the bias-corrected second moment vector $\hat{u}_2$ is calculated

$$\hat{u}_2^T = u_2^T/(1-\beta_2^T). \tag{11}$$

Finally, the parameters are updated for the training of next timestep

$$p^T = p^{T-1} - \alpha\frac{\hat{u}_1^T}{\sqrt{\hat{u}_2^T}+\epsilon}. \tag{12}$$

In practice, β_1 is set to be 0.9, β_2 is set to be 0.999, ϵ is set to be 10^{-8}, and the learning rate α is set to be 0.001 as suggested in [89].

In order to further improve the training efficiency and robustness, the pretraining and fine-tuning method [90], [91] is employed. Different from training with randomly initialized parameters (w and b), fine-tuning initializes using parameters that are pretrained on very large dataset, such as ImageNet [92] or COCO [93]. Therefore, these pretrained parameters are expected to be capable of identifying numerous image patterns or features for common objects. As a result, fine-tuning is particularly useful for applications that have small training data sets [94].

In this work, the unmodified front part of the attention model is initialized using the COCO pretrained Tiny-YOLOv3 parameters while the modified parts are initialized using the HE method. Then, the pretrained parameters of the attention model are frozen, and the remaining parts of the attention model are trained using the training data in this work. As recommended in [64], [76] and the document files of the YOLO projects, this training is performed using the MBGD and the ADAM optimizer with a relatively large learning rate, 0.001. Once the learning error curve converges in about 50 epochs, all trainable parameters of the attention model are unfrozen and fine-tuned using the same process but a smaller learning rate, 0.0001, for 50 epochs.

C. Fine-grained localization method based on ITMM

With the ROIs predicted from the attention model, the ITMM is employed to perform fine-grained localization for the points of interest. Template matching is a classical computer vision method for localization of objects [37], [95]. Due to its simplicity, explainability, and ease of deployment, it is widely employed for a lot of applications, such as object detection, tracking, and image stitching. The process of template matching method involves two major components: the source image and the template patch. The template is a small image (H:40 pixels, W:30 pixels in this work) patch that clearly defines the object to be detected in the much larger source image (H:$\sim$100 pixels, W:$\sim$100 pixels in this work). In this work, the source images are the ROIs predicted from the attention model, and the template is obtained by averaging patches of object pole tips in the training dataset. The pole tip/joint point to be welded is located at the geometry center of the template patch. Both the source images and the template patch are converted into gray-scale for convenience of computation. In the source image, the template patch slides one pixel each time and calculates a value of similarity between the template patch and the overlapped source patch. In this work, the value of similarity is defined as the Normalized Cross-Correlation (NCC):

$$\begin{aligned}&\mathrm{NCC}(\mathrm{patch_s}, \mathrm{patch_t}) = \\ &\frac{1}{N\sigma_s\sigma_t}\sum(\mathrm{patch_s} - \overline{\mathrm{patch_s}}) \times (\mathrm{patch_t} - \overline{\mathrm{patch_t}}),\end{aligned} \tag{13}$$

where N is number of pixels in the patch, σ is the standard deviation of patch pixel intensities, and the subscripts s and t represent the source image patch and the template patch, respectively, $\overline{\mathrm{patch}}$ is the average pixel intensity of the patch.

The template matching will return to an NCC matrix, which will be smaller than the source image. For example, if the size of the source image is $M_s \times N_s$ and the size of the template is $M_t \times N_t$. Then, the size of the NCC matrix will be $(M_s - M_t + 1) \times (N_s - N_t + 1)$. Theoretically there is only one welding joint point in each ROI detected by the attention model. As a result, the position with the maximum NCC in the NCC matrix will be identified as the matching position. However, localization based on the maximum NCC may be sensitive to unexpected image noise. Therefore, a 2×2 average pooling filter is applied to the NCC matrix for more robust localization performance. The element with the highest value in the filtered matrix will be found and its corresponding coordinates in the source image will be calculated as the object location.

Scale invariance is one of the greatest challenges to the ITMM. Changes in the size or orientation of the objects in the source image will adversely affect the accuracy of the algorithm [37]. Fortunately, the objects, such as the metal sleeves and the shelf frames, are firmly hold in fixed orientations by the fixtures, and the attention model will exclude the

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388

noisy backgrounds. Other potential fine-grained localization methods, such as shape fitting [38], edge-based matching [39], and Scale-Invariant Feature Transform (SIFT) [40], are investigated. However, these advanced methods do not show significantly better performance in term of either accuracy or computational efficiency when validated using the training data. Therefore, the ITMM is selected in this work as the fine-grained localization model due to its simplicity, consistency, and extremely high precision [96].

D. Method summary

In summary, the training and the inference methods of the FGAM are presented in Algorithm 1 and Algorithm 2, respectively. With the deep learning based attention model, the proposed FGAM first detects ROIs to minimize the background noise. The backbone of the attention model is optimized based on Tiny-YOLOv3 for fast computation. Followed by an ITMM, the FGAM then precisely localizes the points of interest without being adversely affected by the ROI jittering. Together with multiple optimization strategies, such as modifications in network architecture, initialization with pretrained weights, average pooling filter for NCC matrix, the FGAM can be fine-tuned for specific tasks using a small dataset. Therefore, the proposed FGAM is able to achieve fast, highly accurate, and robust vision localization for robot guidance in dynamic manufacturing environments.

In a dynamic environment, the conventional deep learning models have to be frequently retrained and optimized in order to accurately identify new types of defects. In comparison, the hybrid learning method detects new types of defects with a relatively satisfactory accuracy. Therefore, the efforts of iterative model optimization and corresponding maintenance costs can be notably reduced.

Algorithm 1 FGAM training

1: Build the attention model (the modified Tiny-YOLOv3);
2: Initialize the trainable parameters of the attention model, the unmodified Tiny-YOLOv3 layers are initialized using the pretrained COCO parameters and the modified layers are initialized using the HE method;
3: Set the loss function as YOLO loss;
4: Fine-tune the modified layers of the attention model based on the training data $\{x_i, (y^j_{\mathrm{h},i}, y^j_{\mathrm{w},i})\}$ using the MBGD and the ADAM optimizer with a relatively large learning rate of 0.001;
5: Fine-tune the all trainable parameters of the attention model based on the same training data using the MBGD and the ADAM optimizer with a smaller learning rate of 0.0001;
6: Develop the ITMM model and prepare the template by averaging patches (H:40 pixels, W:30 pixels) of the object pole tips in the training dataset;
7: Attach the ITMM model subsequent to the attention model and therefore form the FGAM.

Algorithm 2 FGAM inference

Input: New captured image x_{new}, NMS IoU threshold θ, the ITMM template $\mathrm{patch_t}$
Output: Localization prediction $\hat{y}^j_{\mathrm{h,new}}, \hat{y}^j_{\mathrm{w,new}}$

1: Input a new image x_{new} to the attention model and generate the ROIs $\hat{v}^j_{\mathrm{new}}$ based on θ;
2: Calculate the NCC matrix using the $\hat{v}^j_{\mathrm{new}}$ ($\mathrm{patch_s}$) and the $\mathrm{patch_t}$;
3: Apply a 2×2 average pooling filter to the NCC matrix;
4: Find the coordinate of the largest element in the filtered matrix and derive its corresponding coordinate $(\hat{y}^j_{\mathrm{h,new}}, \hat{y}^j_{\mathrm{w,new}})$ in the original image x_{new}.

V. Experiments and Results

The FGAM based edge computing system is deployed in a small-scale but representative factory for welding robot guidance. Only 159 images are provided to train and validate the proposed model. The performance of the FGAM localization is assessed using metrics in terms of precision and recall with modified definitions and is compared against several reference models. With the validated FGAM model, the edge IoT enhanced robotic system operates in real-time to mass produce the shelf products. Details of the experiment are presented as follows.

A. Experiment setup

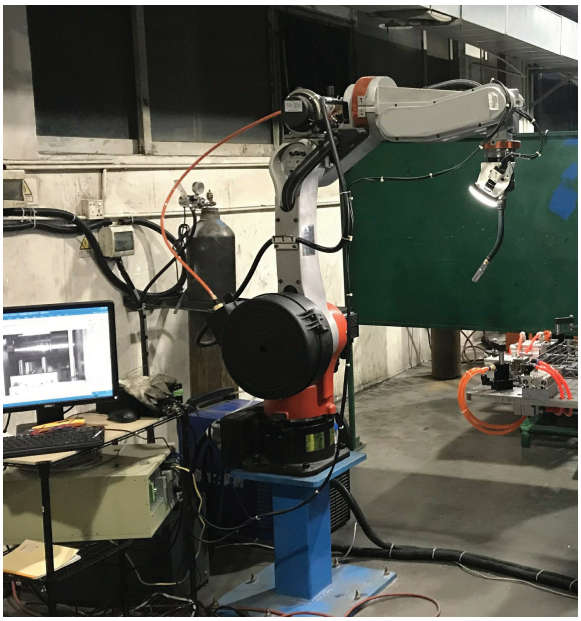
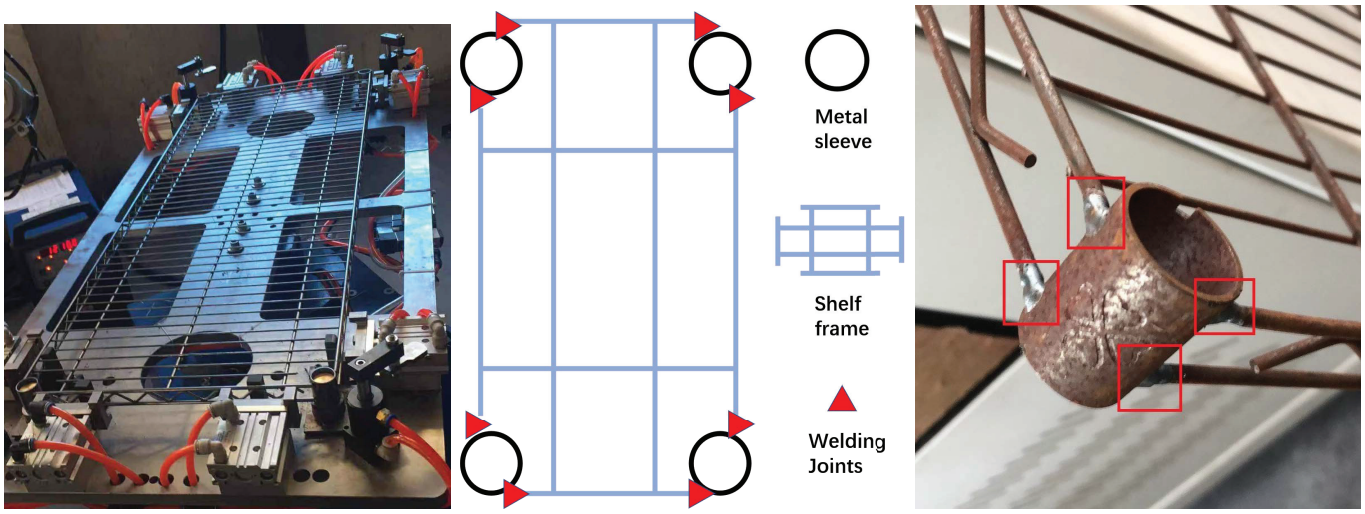


Fig. 3. The robotic welding station.

In this work, the FGAM enhanced edge computing system is deployed on an automatic robotic welding station in a manufacturing factory located in Guangdong, China. A picture of the welding robot station is presented in Fig.3, and details of the employed system are described as follows.

- The edge computing server uses an I7-8700 CPU, a 32G RAM, 128GB SSD, and 1T hard disk drive. Essential

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388



(a) An assembled metal shelf after the welding process.

(b) The schematic diagram of the product (view from top).

(c) A cylindric metal sleeve and welding beads at the corner.

(d) A representative image captured by the vision system.

(e) Representative images of OK weld beads.

(f) Representative images of NG weld beads due to weld deviations.

Fig. 4. Images of the products and the weld beads.

software installed in this server includes Linux 16.04 and Python 3.6. The edge computing server communicates with the vision system, the robot controller, and the user interface device through standard TCP protocols. The edge computing server controls the overall operation.

- The vision system uses a MV-CA020-20GM area scan camera and a Rsee P-HRL ring-shape LED work light. The camera captures 1024×1280 BMP images with an exposure time of 100ms and forwards images to the edge computing server through an ethernet cable. Both the camera and work light are protected by a replaceable glass dome from welding spatters or sparks. The vision system can be either scene-related or object-related [18]. In this work, the vision system is mounted on the robot manipulator (eye-in-hand) and therefore is in a typical object-related scenario.
- The welding robot system contains a 6-axis 550kg robot arm, a 220V, 120A arc welder, and a rotary welding positioner with fixtures. The robot arm has 2239mm reach, 50kg payload, and ±0.06mm repeatability.
- The user interface device uses a 9-inch industrial-grade touch screen monitor. Operators can initialize or reboot the system, set up operation configurations, or check production statistics using this device.

This system is applied to assemble steel wire shelves (shown in Fig. 4(a)) using hollow cylindrical metal sleeves and the shelf frames. An assembled metal shelf has a rectangular shape and has 4 metal sleeves at every corner. A schematic diagram of the parts is presented in Fig. 4(b). For each corner, the metal sleeve joins the frame poles on two sides, and each side will have two spot welding joints (shown in Fig. 4(c) marked in red boxes). Therefore, there are 16 joint points to be welded for each final product. The diameter of the shelf poles is about 5mm, and the wall thickness of the cylindrical metal sleeves is about 2mm.

It takes about 2-4 weeks to set up and calibrate the system until it is ready for mass production. In operation, the assembly process includes five steps: First, four cylindrical metal sleeves and one shelf frame are held tightly by the fixtures on the welding positioner. Second, the robot arm carries the vision system to capture images of joints to be welded. A representative captured image is presented in Fig. 4(d) and the joints to be welded in the image are marked in yellow boxes. Third, the images are received and analyzed by the edge computing server, and the image position coordinates of the joint points are derived by the embedded FGAM model. Fourth, the image position coordinates are converted into the robot coordinates and are sent to the robot arm, which will then use a welding gun to perform the welding operation based on the coordinates. Similar procedures will be repeated until all

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388

four metal sleeves are attached to the shelf frame.

The quality of the final product is inspected by human inspectors, who visually examine the appearance of weld beads. Other inspection methods, such as radiographic or ultrasonic tests, are not employed due to the considerations of cost and efficiency. Therefore, the human visual inspection result is used as the ground truth for both model training and validation in this work. For convenience, the human inspectors denote qualified weld beads as OK and defective weld beads as Not Good (NG). Examples of OK and NG weld beads are presented in Fig. 4(e) and Fig. 4(f), respectively. Therefore, 1% NG rate for weld beads will result in about 15% NG rate for the final shelf product.

Traditionally, the welding assembly is conducted by skilled welding workers and the assembly of each shelf takes about 40 seconds. However, the welding arc and fume are hazardous to human health even with appropriate protections. In addition, labor shortage of skilled workers sometimes even halts the production. Therefore, this factory is in eager to look for automatic welding solutions. Previously, a welding robot without any vision guidance has been employed in this factory. Nevertheless, these welding robots suffer from an overall defect rate around 15% mainly due to weld deviations (exemplified in Fig. 4(f)). Therefore, this factory demands vision guidance solutions that could substantially reduce the defect rate. To this end, a 3-D laser seam tracking method has been considered for this scenario but eventually is not employed because of cost and efficiency. In summary, an applicable vision guidance system in this manufacturing scenario should satisfy the following technique requirements. 1) Low-cost: only 2-D camera solution is within the budget. 2) High efficiency: the customer demands that the total time to produce each product should be less than 1 minute in order to maintain the cost-efficiency of the robot arm. Consequently, after excluding the time needed for material loading, robot arm movement, and welding operation, the vision localization process for all 16 joint points of each product should be completed in 3 seconds, which is about 200ms for one individual joint point. 3) High accuracy and robustness: any localization deviation larger than 1mm may result in NG weld beads, and a final product will be identified as defective if any of the 16 weld beads is identified as NG.

B. Experiment data

In this work, only 159 images are provided by the factory to develop the proposed and all reference models because of many restrictions, such as safety, security, and costs. A representative image from the vision system is presented in Fig. 5. In this image, the example object ROIs and the fine-grained localization points of interest are marked by yellow boxes and yellow dots, respectively. The ground-truth ROIs and the fine-grained localization point positions are both manually labelled for each image. Labelled ROIs are used to develop the attention model, and the labelled fine-grained points of interest are used to validate the performance of the final localization. These labelled images are randomly split into a set of 100 images for training and a set of 59 images for testing.

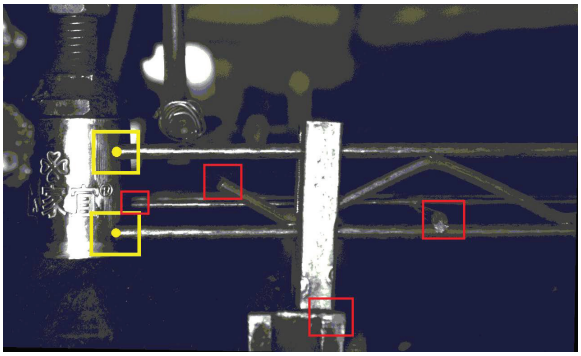


Fig. 5. A representative image captured by the vision system. The welding joint positions to be localized are marked by yellow dots. The ROIs of welding joints are marked by the yellow boxes. The positions that are not welding joints but have similar visual appearances, which might disturb the vision localization, are marked by red boxes.

To train and optimize the attention model with a very small dataset, data augmentation [97], [98] is used to increase the number and diversity of data. Data augmentation enlarges the training dataset by generating modified or synthetic copies of the existing data. Therefore, new data from data augmentation is useful to reduce overfitting and to improve both performance and robustness of the trained model. In this work, employed data augmentation techniques include shearing, zooming, horizontal and vertical flipping, brightness changing, and noise adding.

In order to further enhance the robustness of the localization model, the pixel intensities of the images are projected from 0-255 to 0-1 using minmax normalization method [99]:

$$I_{\text{normalized}} = \frac{I - I_{\text{min}}}{I_{\text{max}} - I_{\text{min}}}, \tag{14}$$

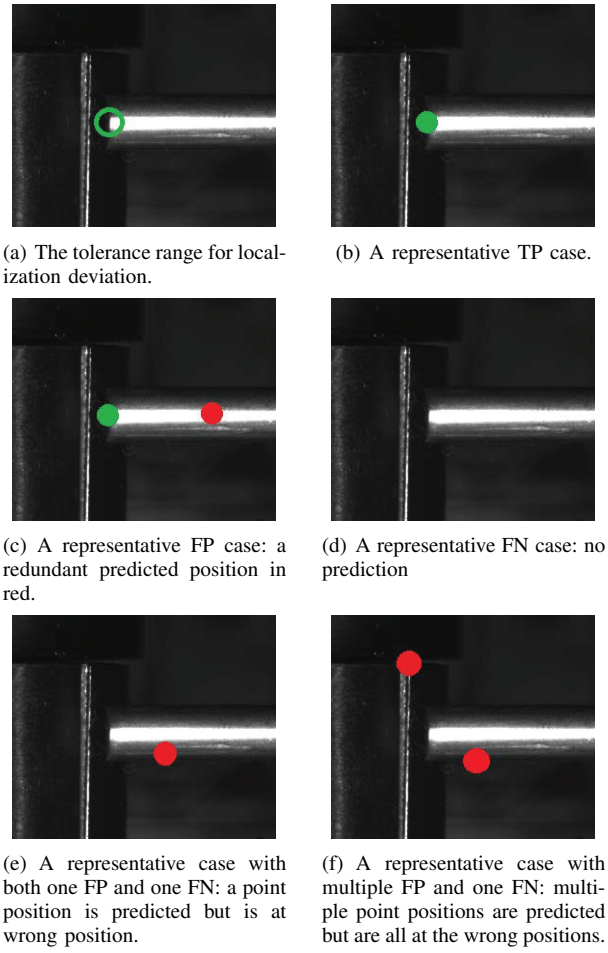
where I is the intensity of every pixels, I_{min} and I_{max} are the minimum (0) and the maximum (255) pixel intensities, respectively.

C. Evaluation metrics

The mean Average Precision (mAP) is a popular metric in measuring the accuracy of object detectors like Faster R-CNN, SSD, etc. The mAP computes the average precision value for recall value over 0 to 1 based on the IoU of predicted and ground-truth bounding boxes. However, the mAP is inappropriate to the application scenario in this work because of the following two reasons:

- The robot guidance requires point coordinates of the welding joints as inputs instead of bounding boxes. The localization accuracy of the predicted point position instead of the IoUs determines whether the welding bead is OK or NG.
- Identifying a nonexistent position to be weld is a serious issue that may be dangerous to the operation safety. The mAP metric does not explicitly evaluate the frequency of the wrong predictions that may cause damage to the system hardware.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388



(a) The tolerance range for localization deviation.

(b) A representative TP case.

(c) A representative FP case: a redundant predicted position in red.

(d) A representative FN case: no prediction

(e) A representative case with both one FP and one FN: a point position is predicted but is at wrong position.

(f) A representative case with multiple FP and one FN: multiple point positions are predicted but are all at the wrong positions.

Fig. 6. Examples of TP, FP, and FN cases.

Metrics to assess the localization performance should be meaningful and associated to the final quality inspection requirements of the welding beads. Therefore, we use the metrics of precision and recall based on the recommendations of the quality management group in the factory. Precision and recall are calculated based on metrics in terms of True Positive (TP), False Positive (FP), False Negative (FN) [100]. The precision is defined as the correctness of positive localizations:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\%. \tag{15}$$

The recall is defined as the effectiveness of localization. In other words, the correctness of localizations for all ground truth positions:

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\%. \tag{16}$$

TP, FP, FN are defined according to the quality inspection requirements in this work. According to the interview of the quality management team, the position of predicted point should be within 1mm range of the mid points of the pole tip (shown in Fig. 6(a)). 1mm is approximately equal to the length of 6 or 7 pixels in a standard capture image. Therefore, TP is defined as that the position of labelled point is correctly predicted (shown in Fig. 6(b)). In other words, the distance between the predicted point position and the labelled point position is less or equal to 7 pixels. FP is defined as that a predicted position is generated but there is no labelled position within the 7-pixel distance. For example, a redundant point is generated as shown in Fig. 6(c). FN is defined as that a labelled position is not predicted (as shown in Fig. 6(d)). If an incorrect position is predicted (as shown in Fig. 6(e)), values of both FP and FN will increase by 1. If multiple predicted positions are generated incorrectly (as shown in Fig. 6(f)), values of FP will increase by the number of incorrect predictions but values of FN will still increase by 1.

For any employed localization model, both precision and recall should be very high to maintain a low defective rate. In addition, the quality management group concerns more about the FP case that might be dangerous to the system and human operators. Consequently, precision rate nearly equal to 1 is a mandatory requirement for automatic localization models. Therefore, a few more mechanisms are employed to minimize the probability of FP cases: 1) the number of detected ROIs from the attention model is set to 2; 2) discard all ROIs that have confidence scores less than 0.85; 3) for each detected ROI, the fine-grained localization model generates only one prediction based on filtered NCC matrix.

D. Reference models

In this work, five reference models are developed to benchmark the performance of the proposed FGAM. The first reference model employs only the conventional computer vision method (the ITMM), which has been described in Section IV-C. The second reference model is the original Tiny-YOLOv3 detector, and the third reference model is the modified Tiny-YOLOv3 detector without the fine-grained enhancement. For convenience of comparison, the second and the third reference detectors are trained using the same strategy, such as fine-tuning and ADAM optimizer that are described in Section IV-B. However, the robot guidance requires point positions instead of bounding boxes. Therefore, the geometry center ($\hat{b}_{\text{h}}$ and $\hat{b}_{\text{w}}$) of each ROI box is used as the output of the reference detector. The fourth reference model is the Single Shot multibox Detector (SSD) [60], which is another representative one-stage object detector. SSD demonstrates high detection efficiency and is potential for the application scenario. The last reference model is a deep CNN based object localization model, which directly builds the mathematical relationship from the input image to the point coordinates of object points. Different from the backbone of the reference Tiny-YOLOv3 detector, this localization model employs relatively deeper CNN architecture, which is the VGG network [101] proposed by the Visual Geometry Group at University of Oxford. The VGG network has relatively deep structure using small (3×3 and 1×1) convolution filters to extract high-level representations. To convert the VGG into a localization model, the following processes are employed. First, the original predefined classifier of the original VGG network is removed and only the front CNN part is left as an image feature extractor. A fully connected network based localizer is attached to the VGG feature extractor. This network contains three Fully Connected

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388

(FC) layers. The first FC layer has 128 neurons for image feature abstraction. The second FC layer has 32 neurons for further refinement of features. Each captured image has two welding joint positions with 4 coordinates (y_{h}^1, y_{w}^1, y_{h}^2, y_{w}^2) to be predicted. Therefore, the last FC layer has 4 neurons and the 4 outputs of the neurons are corresponding to the 4 coordinates.

E. Performance of the fine-grained attention model

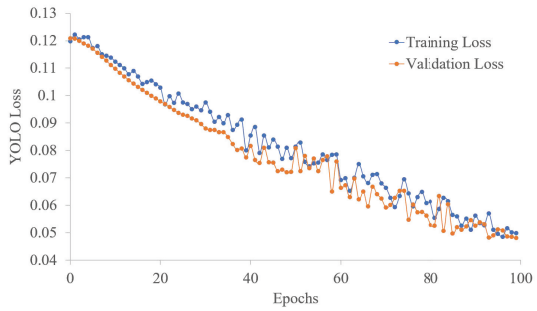


Fig. 7. The curves of both training and validation losses with training epochs.

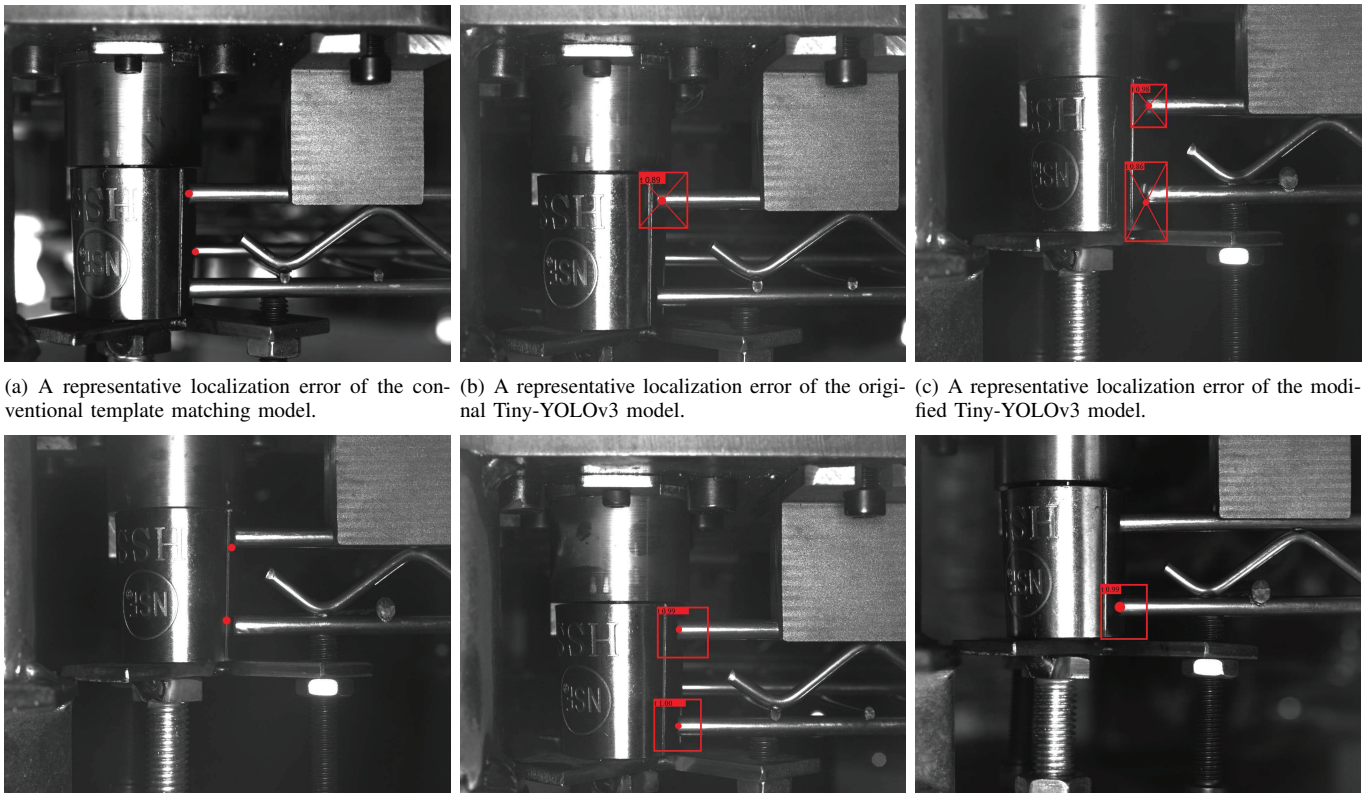
The proposed fine-grained attention model as well as the 4 reference models are assessed on the testing data set defined in Section V-B. We first evaluate the curves of both training and validation losses, which are presented in Fig. 7. There is no significant gap between the two curves. Therefore, the attention model is less likely to be overfitted after the data-driven training. The assessment results are presented in Table I. The proposed FGAM model significantly outperforms all the reference models in terms of both precision and recall metrics. For instance, the FGAM has only 1 FN case resulting in a precision of 1.00 and a recall over 0.99. Therefore, the FGAM is the only model that potentially satisfies the strict requirements of localization accuracy in this application scenario. The ITMM model has 18 FP and 18 FN cases resulting in a precision of 0.85 and a recall of 0.85, respectively. The modified Tiny-YOLOv3 model has achieved superior performance over the original Tiny-YOLOv3 model, particularly in terms of recall (0.91 to 0.83). Therefore, this result validates the significance of the modified Tiny-YOLOv3 architecture. The reference SSD model demonstrates similar performance as the original Tiny-YOLOv3 model. The recall of SSD model, 0.86, is slightly higher than that of the original Tiny-YOLOv3 model, 0.83, while the precision of SSD model, 0.84, is slightly lower than that of the original Tiny-YOLOv3 model, 0.89). The VGG localization model is suffered from both high FP and FN cases resulting in relatively low precision, 0.45, and recall, 0.45, which suggests that training a deeper model with a relatively small training set may lead to over-fitting issues resulting in lower test accuracy [102].

To further understand the localization performance of all evaluated models, the example images of the localization results are presented in Fig. 8. The conventional template matching model is vulnerable to background noise. When a background pole that has similar visual features appears in the image, the ITMM model might predict the wrong pole tip that should not be welded (shown in Fig. 8(a)). In addition, the robot arm guided by incorrectly predicted positions might damage the system hardware. Therefore, visual guidance based on only the conventional method is not suitable to this application. Deep learning based object detection models predict bounding boxes for ROIs of the welding joints. Therefore, they are able to distinguish the joint positions to be welded from background noise. However, the original Tiny-YOLOv3 has relatively high probability of FN cases with a recall of only 0.83. An FN example of the original Tiny-YOLOv3 is presented in Fig. 8(b). With modified architecture that is specifically designed for this work, the modified Tiny-YOLOv3 significantly reduces the FN ratios. However, the geometry center of a predicted bounding box is not necessarily located exactly at the pole tip position (shown in Fig. 8(c), the lower red box) because of the bounding box jittering. The deviations between the geometry centers of bound boxes and the labelled positions may exceed the 1-mm limitation and result in defective welding beads. The VGG based deep localization model is employed to directly predict the point positions to be welded. However, the proximity of the VGG predicted positions to the true joint positions is varying even more seriously than that of modified Tiny-YOLO-v3 (shown in Fig. 8(d)).

Therefore, the FGAM has shown substantial advantages in localizing the points of interest. With ROIs predicted from the attention model, the FGAM is robust against the background noise (shown in Fig. 8(e)). Within the predicted ROIs, the fine-grained ITMM model is employed to localize the precise positions of welding joint points. Therefore, the deviations between the final predicted positions and the ground-truth positions can be minimized even when the ROIs from the attention model are jittering (shown in Fig. 8(e), the upper red box). The only FN case happens when the attention model predicts only one ROI as shown in Fig. 8(f). Fortunately, this problem will not damage the system hardware and can be solved by sending alarm for manual intervention.

After the validation using historical images, FGAM is compiled and installed on the edge computing system (described in Section V-A). Then, the edge system is mounted on the welding robot to operate in real-time for mass production in the cooperated factory. The operational data is no longer available due to the security regulations of this factory. After one week of mass production, the quality management group reports that this robotic welding system has produced about 6000 products. The defective rate is about 5% due to a variety of issues, such as inappropriate welding configurations, exhaustion of protection gas, electrical faults, and automation errors. Only 2 defects are identified to be related to the vision localization errors. These 2 localization errors are FN cases similar to that in Fig. 8(f)), which may be caused by overexposure problems (e.g. sun light, light of forklift truck, inappropriate reflection of work light) as suggested by the production operators. Therefore, the vision localization accuracy of the FGAM is approximately 99.96% for one product and 99.998% for each individual welding joint.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388



(a) A representative localization error of the conventional template matching model.

(b) A representative localization error of the original Tiny-YOLOv3 model.

(c) A representative localization error of the modified Tiny-YOLOv3 model.

(d) A representative localization error of the deep VGG model.

(e) A representative image of correct FGAM localizations.

(f) A representative localization error of the FGAM.

Fig. 8. Representative images of localization errors

TABLE I: The assessment performance of the proposed fine-grained attention model and the reference models.

Models	TP	FP	FN	Precision	Recall	Inference time
ITMM model	100	18	18	0.85	0.85	20-30 ms
Original Tiny-YOLOv3	98	12	20	0.89	0.83	40-50 ms
Modified Tiny-YOLOv3	107	10	11	0.91	0.91	20-30 ms
SSD	102	20	16	0.84	0.86	100-120 ms
VGG model	53	65	65	0.45	0.45	60-70 ms
FGAM	117	0	1	1.00	0.99	60-70 ms

F. Future Research Directions

In future work, there are a few approaches we could adapt to both validate the generalization of the method and further improve the overall performance of the system.

First, in addition to robot guidance, effort will be spent on generalizing the proposed method to other smart manufacturing scenarios. This work aims to develop a computer vision model on isolated edge computing system in IoT of smart manufacturing scenario. Many manufacturing scenarios usually have similar challenges (e.g. small data, strict accuracy, efficiency, costs). Therefore, the proposed method is highly potential to be generalized to new vision localization/detection scenarios, which include but not limited to material loading, missing part detection, production line sorting, and product counting.

Second, the FGAM will be optimized to further improve its performance and robustness. State-of-the-art neural networks, such as ViT based YOLO [103], will be analyzed and compared with both the current attention model and the fine-grained localization model. We will actively search for detection approaches that are capable of addressing the jittering issues. Once the jittering issue is solved, the FGAM can be simplified to an end-to-end approach, which will both reduce the implementation costs and increase the operational robustness.

Third, the hyper-parameters could be further optimized based on collected data of the scenario. It is impractical to employ exhaustive search methods because the search space for hyper-parameters is inexhaustible. Genetic algorithm [104] can be considered as a potential approach to further optimize the hyper-parameters.

Last but not least, Auto Machine Learning (AutoML) [105] can be employed to periodically upgrade the deployed model during operation. Both hardware and software of the coal fired plants might be subjected to changes due to upgrading or maintenance purpose. Therefore, instead of manually collecting new data and retraining the model after system changes, fine tuning a data driven model periodically will substantially enhance the model robustness and decrease the maintenance

costs.

VI. Conclusion

In this work, a low-cost edge computing IoT system is developed for vision guidance of welding robots in smart manufacturing scenarios. To predict highly accurate positions of target objects in long-term mass production with high computing efficiency, a Fine-Grained Attention Model (FGAM) is proposed for the object-related vision system of the proposed IoT system. The proposed FGAM first uses a modified Tiny-YOLOv3 to predict ROIs from input images and then uses an improved template matching model to perform fine-grained localization in the predicted ROIs. With very limited data for training, enhancement methods, such as data augmentation, fine-tuning with pretrained parameters, mini batch gradient descent, ADAM optimizer, and average pooling filter, are employed to improve the performance and the generalization of the model. Validated on in-field data collected from the IoT vision system, the proposed FGAM significantly outperforms all the reference models in terms of both precision (1.0) and recall (0.99). Then, the FGAM based IoT system is deployed for mass production in real-time achieving an efficient processing and transmission rate of 20 images per second. In nearly 96000 vision guided welding operations, only 2 false negative localization errors are identified achieving a localization recall up to 99.998%. The proposed FGAM is expected to noticeably enhance the adaptability, cost-efficiency, and market acceptance of IoT systems for robot guidance applications. This work is particularly relevant to automatic operation and quality control for manufacturing industry.

Acknowledgements

The authors gratefully acknowledge the support provided by Shiwei Qin, Shuai Dai, Shuyi Wang, Yingjie He, and Siqi Huang who assisted in the data collections, experiment coordinates, hardware preparation, system and software deployment, and research discussions. The authors also gratefully acknowledge the laboratory and platform support from AIATOR Co., Ltd. and Googoltech Group and the technique advice from Xiaogang Xiong, Shu lv, and Hong Wu.

References

[1] T. C. Chiu, Y. Y. Shih, A. C. Pang, C. S. Wang, W. Weng, and C. T. Chou, "Semisupervised distributed learning with non-iid data for AIoT service platform," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9266–9277, 2020.

[2] H. Wu, Z. Zhang, C. Guan, K. Wolter, and M. Xu, "Collaborate edge and cloud computing with distributed deep learning for smart city internet of things," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8099–8110, 2020.

[3] H. S. Kang, J. Y. Lee, S. Choi, H. Kim, J. H. Park, J. Y. Son, B. H. Kim, and S. Do Noh, "Smart manufacturing: Past research, present findings, and future directions," *Int. J. Precision Eng. Manuf.-green Technol.*, vol. 3, no. 1, pp. 111–128, 2016.

[4] Y. Zhang, J. Ren, J. Liu, C. Xu, H. Guo, and Y. Liu, "A survey on emerging computing paradigms for big data," *Chinese J. Electron.*, vol. 26, no. 1, pp. 1–12, 2017.

[5] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the edge: A scalable IoT architecture based on transparent computing," *IEEE Netw.*, vol. 31, no. 5, pp. 96–105, 2017.

[6] Z. Meng, Z. Wu, and J. Gray, "RFID-based object-centric data management framework for smart manufacturing applications," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2706–2716, 2019.

[7] A. Kusiak, "Smart manufacturing," *Int. J. Prod. Research*, vol. 56, no. 1-2, pp. 508–517, 2018.

[8] H. Tang, D. Li, J. Wan, M. Imran, and M. Shoaib, "A reconfigurable method for intelligent manufacturing based on industrial cloud and edge intelligence," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4248–4259, 2020.

[9] B. Esmaeilian, S. Behdad, and B. Wang, "The evolution and future of manufacturing: A review," *J. Manuf. Syst.*, vol. 39, pp. 79–100, 2016.

[10] F. Tao, Q. Qi, A. Liu, and A. Kusiak, "Data-driven smart manufacturing," *J. Manuf. Syst.*, vol. 48, pp. 157–169, 2018.

[11] Z. M. Fadlullah, F. Tang, B. Mao, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems," *IEEE Commun. Surv. Tutor.*, vol. 19, no. 4, pp. 2432–2455, 2017.

[12] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *J. Manuf. Syst.*, vol. 48, pp. 144–156, 2018.

[13] S. Verma, Y. Kawamoto, Z. M. Fadlullah, H. Nishiyama, and N. Kato, "A survey on network methodologies for real-time analytics of massive IoT data and open research issues," *IEEE Commun. Surv. Tutor.*, vol. 19, no. 3, pp. 1457–1477, 2017.

[14] S. Chen, H. Xu, D. Liu, B. Hu, and H. Wang, "A vision of IoT: Applications, challenges, and opportunities with china perspective," *IEEE Internet Things J.*, vol. 1, no. 4, pp. 349–359, 2014.

[15] L. Pérez, Í. Rodríguez, N. Rodríguez, R. Usamentiaga, and D. F. García, "Robot guidance using machine vision techniques in industrial environments: A comparative review," *Sensors*, vol. 16, no. 3, p. 335, 2016.

[16] J. L. Sanz, *Advances in Machine Vision*. Springer Science & Business Media, 2012.

[17] C. Wöhler, *3D Computer Vision: Efficient Methods and Applications*. Springer Science & Business Media, 2012.

[18] G. Alenyà, S. Foix, and C. Torras, "Tof cameras for active vision in robotics," *Sens. Actuators, A*, vol. 218, pp. 10–22, 2014.

[19] H. Li, K. Ota, and M. Dong, "Learning IoT in edge: Deep learning for the internet of things with edge computing," *IEEE Netw.*, vol. 32, no. 1, pp. 96–101, 2018.

[20] J. Liu, H. Guo, H. Nishiyama, H. Ujikawa, K. Suzuki, and N. Kato, "New perspectives on future smart FiWi networks: Scalability, reliability, and energy efficiency," *IEEE Commun. Surv. Tutor.*, vol. 18, no. 2, pp. 1045–1072, 2015.

[21] S. R. Pokhrel, L. Pan, N. Kumar, R. Doss, and H. L. Vu, "Multipath tcp meets transfer learning: A novel edge-based learning for industrial iot," *IEEE Internet Things J.*, vol. 8, no. 13, pp. 10 299–10 307, 2021.

[22] S. Singh and N. Singh, "Internet of things (IoT): Security challenges, business opportunities & reference architecture for e-commerce," in *Proc. Int. Conf. Green Comput. Internet Things (ICGCIoT)*, 2015, pp. 1577–1581.

[23] J. Zhang, B. Chen, Y. Zhao, X. Cheng, and F. Hu, "Data security and privacy-preserving in edge computing paradigm: Survey and open issues," *IEEE Access*, vol. 6, pp. 18 209–18 237, 2018.

[24] R. Hadidi, J. Cao, M. S. Ryoo, and H. Kim, "Toward collaborative inferencing of deep neural networks on internet-of-things devices," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 4950–4960, 2020.

[25] Y. Chu, D. Feng, Z. Liu, Z. Zhao, Z. Wang, X.-G. Xia, and T. Q. S. Quek, "Hybrid learning based operational visual quality inspection for edge computing enabled iot system," *IEEE Internet Things J.*, pp. 1–1, 2021.

[26] F. Yu, L. Cui, P. Wang, C. Han, R. Huang, and X. Huang, "Easiedge: A novel global deep neural networks pruning method for efficient edge computing," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1259–1271, 2021.

[27] T. Clarke and X. Wang, "The control of a robot end-effector using photogrammetry," *ISPRS J. Photogramm. Remote Sens.*, vol. 33, no. B5/1; PART 5, pp. 137–142, 2000.

[28] Y. Zhao, K. Xu, H. Wang, B. Li, M. Qiao, and H. Shi, "Mec-enabled hierarchical emotion recognition and perturbation-aware defense in smart cities," *IEEE Internet Things J.*, vol. 8, no. 23, pp. 16 933–16 945, 2021.

[29] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE T. Neur. Net. Lear.*, vol. 30, no. 11, pp. 3212–3232, 2019.

[30] H. Zhang and N. Wang, "On the stability of video detection and tracking," *arXiv preprint arXiv:1611.06467*, 2016.

[31] B. Zhao, J. Feng, X. Wu, and S. Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *Int. J. Control Autom.*, vol. 14, no. 2, pp. 119–135, 2017.

[32] E. Gavves, B. Fernando, C. G. Snoek, A. W. Smeulders, and T. Tuytelaars, "Fine-grained categorization by alignments," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1713–1720.

[33] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based R-CNNs for fine-grained category detection," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 834–849.

[34] T. Borangiu and A. Dumitrache, *Robot Arms with 3D Vision Capabilities*. INTECH Open Access Publisher, 2010.

[35] Z. Liu, Y. Xie, J. Xu, and K. Chen, "Laser tracker based robotic assembly system for large scale peg-hole parts," in *Proc. IEEE 4th Annu. Int. Conf. Cyber Technol. Autom. Control Intell. Syst. (CYBER)*. IEEE, 2014, pp. 574–578.

[36] R. Cipolla and N. J. Hollinghurst, "Visual robot guidance from uncalibrated stereo," *Real-time Comput. Vis.*, pp. 169–187, 1994.

[37] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*. John Wiley & Sons, 2009.

[38] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "scikit-image: image processing in Python," *PeerJ*, vol. 2, p. e453, 2014.

[39] M. Chandraker, J. Lim, and D. Kriegman, "Moving in stereo: Efficient structure and motion using lines," in *Proc. IEEE Int. Conf. Comput. Vis.* IEEE, 2009, pp. 1741–1748.

[40] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[41] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2006, pp. 404–417.

[42] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2006, pp. 430–443.

[43] N. O'Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G. V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh, "Deep learning vs. traditional computer vision," in *Proc. Sci. Info. Conf.* Springer, 2019, pp. 128–144.

[44] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Europ. Conf. Comp. Vis. (ECCV)*, 2014, pp. 818–833.

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.

[46] C. W. Chen, "Internet of video things: Next-generation IoT with visual sensors," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 6676–6685, 2020.

[47] D. Weimer, B. Scholz-Reiter, and M. Shpitalni, "Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection," *CIRP Annals*, vol. 65, no. 1, pp. 417–420, 2016.

[48] R. Ren, T. Hung, and K. C. Tan, "A generic deep-learning-based approach for automated surface inspection," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 929–940, 2017.

[49] J. Masci, U. Meier, D. Ciresan, J. Schmidhuber, and G. Fricout, "Steel defect classification with max-pooling convolutional neural networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2012, pp. 1–6.

[50] J.-K. Park, B.-K. Kwon, J.-H. Park, and D.-J. Kang, "Machine learning-based imaging system for surface defect inspection," *Int. J. Precision Eng. Manuf.-Green Technol.*, vol. 3, no. 3, pp. 303–310, 2016.

[51] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013.

[52] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 648–656.

[53] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587.

[54] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, 2015.

[55] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[56] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[57] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

[58] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.

[59] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2147–2154.

[60] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 21–37.

[61] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.

[62] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," *arXiv preprint arXiv:1701.06659*, 2017.

[63] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 740–755.

[64] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[65] ——, "YOLO9000: better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7263–7271.

[66] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.

[67] B. Zhao, J. Feng, X. Wu, and S. Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *Int. J. Autom. Comput.*, vol. 14, no. 2, pp. 119–135, 2017.

[68] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[69] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3156–3164.

[70] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2204–2212.

[71] L. Itti and C. Koch, "Computational modelling of visual attention," *Nat. Rev. Neurosci.*, vol. 2, no. 3, pp. 194–203, 2001.

[72] L.-Q. Chen, X. Xie, X. Fan, W.-Y. Ma, H.-J. Zhang, and H.-Q. Zhou, "A visual attention model for adapting images on small displays," *Multimed. Syst.*, vol. 9, no. 4, pp. 353–364, 2003.

[73] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Machine Learning*, 2015, pp. 2048–2057.

[74] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2553–2561.

[75] D. Gordon, A. Kembhavi, M. Rastegari, J. Redmon, D. Fox, and A. Farhadi, "IQA: Visual question answering in interactive environments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4089–4098.

[76] Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, and Z. Liang, "Apple detection during different growth stages in orchards using the improved YOLO-V3 model," *Comput. Electron. Agr.*, vol. 157, pp. 417–426, 2019.

[77] D. Xiao, F. Shan, Z. Li, B. T. Le, X. Liu, and X. Li, "A target detection model based on improved tiny-YOLOv3 under the environment of mining truck," *IEEE Access*, vol. 7, pp. 123 757–123 764, 2019.

[78] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[79] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, 2015, pp. 1–9.

[80] J. Hosang, R. Benenson, and B. Schiele, "Learning non-maximum suppression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4507–4515.

[81] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 6, pp. 679–698, 1986.

[82] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[83] A. A. M. Al-Saffar, H. Tao, and M. A. Talab, "Review of deep convolution neural network in image classification," in *Proc. Int. Conf. Radar, Antenna, Microw., Electron., Telecommun. (ICRAMET)*, 2017, pp. 26–31.

[84] R. H. Inman, H. T. C. Pedro, and C. F. M. Coimbra, "Solar forecasting methods for renewable energy integration," *Prog. Energy Combust. Sci.(PECS)*, vol. 39, no. 6, pp. 535–576, 2013.

[85] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Int. Conf. Artificial Intell. Stat. (ICAIS)*, 2010, pp. 249–256.

[86] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.

[87] J. Redmon, "Darknet: Open source neural networks in c," http://pjreddie.com/darknet/, 2013–2016.

[88] L. Bottou and O. Bousquet, "The tradeoffs of large scale learning," in *Proc. Adv. Neural Inform. Process. Syst.*, 2008, pp. 161–168.

[89] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[90] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2009.

[91] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inform. Process. Syst.*, 2014, pp. 3320–3328.

[92] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, 2009, pp. 248–255.

[93] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Europ. Conf. Comp. Vis.(ECCV)*, 2014, pp. 740–755.

[94] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. Int. Conf. Artificial Neural Netw.*, 2018, pp. 270–279.

[95] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[96] H. C. van Assen, M. Egmont-Petersen, and J. H. Reiber, "Accurate object localization in gray level images using the center of gravity measure: accuracy versus precision," *IEEE Trans. Image Process.*, vol. 11, no. 12, pp. 1379–1384, 2002.

[97] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 60, pp. 1–48, 2019.

[98] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[99] J.-M. Jo, "Effectiveness of normalization pre-processing of big data to the machine learning performance," *J. Korea Inst. Electron. Commun. Sci.*, vol. 14, no. 3, pp. 547–552, 2019.

[100] F. Provost, "Machine learning from imbalanced data sets 101," in *Proc. AAAI'2000 Workshop Imbalanced Data Sets*, vol. 68, no. 2000, 2000, pp. 1–3.

[101] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–12.

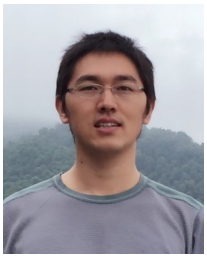
[102] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural comput.*, vol. 29, no. 9, pp. 2352–2449, 2017.

[103] Z. Zhang, X. Lu, G. Cao, Y. Yang, L. Jiao, and F. Liu, "Vit-yolo: Transformer-based yolo for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis*, 2021, pp. 2799–2808.

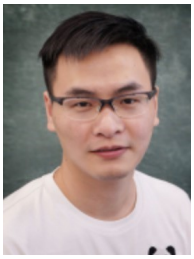
[104] E. Alpaydin, *Introduction to Machine Learning*. MIT Press, 2020.

[105] X. He, K. Zhao, and X. Chu, "Automl: A survey of the state-of-the-art," *Knowledge-Based Systems*, vol. 212, p. 106622, 2021.

Yinghao Chu received his Ph.D. from University of Califnornia, San Diego in 2015. Now Dr. Chu is an assistant professor working at Department of Advanced Design and Systems Engineering, City University of Hong Kong. Dr. Yinghao Chu has been working in the domain of Artificial Intelligence (AI) for real-world application since 2011. His research focuses on hybrid AI, which simulates the attention and coordination mechanism of human intelligence to solve application-orientated problems in real world, particularly in the areas of (1) renewable forecast and application, such as probabilistic forecasts of solar/load time series, and (2) smart manufacturing, such as operational computer vision system for robot guidance and surface quality inspection.



Daquan Feng received his Ph.D. degrees in Information Engineering from the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China in 2015. He had been a visiting student in the School of Electrical and Computer Engineering, Georgia Institute of Technology, USA, from 2011 to 2014. After graduation, he was a research staff in the State Radio Monitoring Center, Beijing, China, and then a Postdoctoral Research Fellow in Singapore University of Technology and Design. He is now an associate professor with the Shenzhen Key Laboratory of Digital Creative Technology, the Guangdong Province Engineering Laboratory for Digital Creative Technology, the Guangdong-Hong Kong Joint Laboratory for Big Data Imaging and Communication, College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China. His research interests include URLLC communications, MEC, and massive IoT networks. Dr. Feng is an Associate Editor of IEEE COMMUNICATIONS LETTERS, Digital Communications and Networks and ICT Express.



Zuozhu Liu , IEEE Member, Assistant professor at Zhejiang University since Sept 2020. Before that, he was a Postdoc Research Fellow at the Department of Statistics and Applied Probability, National University of Singapore. He received his PhD degree from Singapore University of Technology and Design in 2019, and B.Eng from Zhejiang University, College of Information Science and Electrical Engineering. During his undergraduate and PhD studies, he visited and worked at the University of Notre Dame, the AIP Center at RIKEN in Japan, and the AI unicorn Preferred Networks Inc in Tokyo, etc. His research interests mainly lie in machine learning, deep learning algorithms, such as unsupervised/semi-supervised learning algorithms, Bayesian inference, and their applications in healthcare, language representation learning, wireless networks, etc. His research works are published in top journals and conferences such as IEEE JSTSP, IEEE Trans Commun, AAAI, EMNLP, ACL, etc.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3206388

Lei Zhang (Senior Member, IEEE) (Senior Member, IEEE) is a Professor of Trustworthy Systems at the University of Glasgow. He has academia and industry combined research experience on wireless communications and networks, and distributed systems for IoT, blockchain, autonomous systems. His 20 patents are granted/filed in 30+ countries/regions. He published 3 books, and 150+ papers in peer-reviewed journals, conferences and edited books. Dr. Zhang is an associate editor of IoT Journal, IEEE Wireless Communications Letters and Digital Communications and Networks, and a guest editor of IEEE JSAC. He received the IEEE ComSoc TAOS Technical Committee Best Paper Award 2019 and IEEE ICEICT'21 Best Paper Award. Dr. Zhang is the founding Chair of IEEE Special Interest Group on Wireless Blockchain Networks in IEEE Cognitive Networks Technical Committee (TCCN). He delivered tutorials in IEEE ICC'20, IEEE PIMRC'20, IEEE Globecom'21, IEEE VTC'21 Fall, IEEE ICBC'21 and EUSIPCO'21.

Zizhou Zhao received his Bachelor's degree from University of Califnornia, Berkeley in 2015, and his MBA from Cheung Kong Graduate School of Business. He is one of the youngest members of The Chinese Association of Young Scientists and Technologists. He is the founder of AIATOR focusing on industrial Artificial Intelligence (AI) since 2017. His main focus is the improvement of smart manufacture utilizing all the cutting-edge technology such as deep learning, internet of things, industrial robots.

Zhenzhong Wang received the Ph.D. degree in electromagnetic field and microwave technology from Beijing University of Posts and Telecommunications, Beijing, China, in 2010. Since 2010, he has been with the Technology and Management Center of China Central Television which is renamed as China Media Group in 2018. Now, Dr. Wang is a Professorate Senior Engineer. His research interests include 4K/8K UHD production, media contents distribution and 5G transmission.

Zhiyong Feng (M'08-SM'15) received her B.E., M.E., and Ph.D. degrees from Beijing University of Posts and Telecommunications (BUPT), Beijing, China. She is a professor at BUPT, and the director of the Key Laboratory of the Universal Wireless Communications, Ministry of Education, P.R.China. She is a senior member of IEEE, vice chair of the Information and Communication Test Committee of the Chinese Institute of Communications (CIC). Currently, she is serving as Associate Editors-in-Chief for China Communications, and she is a technological advisor for international forum on NGMN. Her main research interests include wireless network architecture design and radio resource management in 5th generation mobile networks (5G), spectrum sensing and dynamic spectrum management in cognitive wireless networks, and universal signal detection and identification.

Xiang-Gen Xia (M'97, S'00, F'09) received his B.S. degree in mathematics from Nanjing Normal University, Nanjing, China, and his M.S. degree in mathematics from Nankai University, Tianjin, China, and his Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, in 1983, 1986, and 1992, respectively.

He was a Senior/Research Staff Member at Hughes Research Laboratories, Malibu, California, during 1995-1996. In September 1996, he joined the Department of Electrical and Computer Engineering, University of Delaware, Newark, Delaware, where he is the Charles Black Evans Professor. His current research interests include space-time coding, MIMO and OFDM systems, digital signal processing, and SAR and ISAR imaging. Dr. Xia is the author of the book Modulated Coding for Intersymbol Interference Channels (New York, Marcel Dekker, 2000).

Dr. Xia received the National Science Foundation (NSF) Faculty Early Career Development (CAREER) Program Award in 1997, the Office of Naval Research (ONR) Young Investigator Award in 1998, and the Outstanding Overseas Young Investigator Award from the National Nature Science Foundation of China in 2001. He received the 2019 Information Theory Outstanding Overseas Chinese Scientist Award, The Information Theory Society of Chinese Institute of Electronics. Dr. Xia has served as an Associate Editor for numerous international journals including IEEE Transactions on Signal Processing, IEEE Transactions on Wireless Communications, IEEE Transactions on Mobile Computing, and IEEE Transactions on Vehicular Technology. Dr. Xia is Technical Program Chair of the Signal Processing Symp., Globecom 2007 in Washington D.C. and the General Co-Chair of ICASSP 2005 in Philadelphia. uished Lecturer of the IEEE Communications Society and a Fellow of IEEE.