CSI2Image: Image Reconstruction From Channel State Information Using Generative Adversarial Networks

SORACHI KATO [1], (Student Member, IEEE), **TAKERU FUKUSHIMA** [2], (Member, IEEE),
TOMOKI MURAKAMI [2], **HIRANTHA ABEYSEKERA** [2], (Member, IEEE), **YUSUKE IWASAKI** [3],
TAKUYA FUJIHASHI [3], (Member, IEEE), **TAKASHI WATANABE** [3], (Member, IEEE),
AND SHUNSUKE SARUWATARI [3], (Member, IEEE)

[1]Division of Electronic and Information Engineering, School of Engineering, Osaka University, Osaka 565-0871, Japan
[2]Access Network Service Systems Laboratories, NTT Corporation, Yokosuka 239-0847, Japan
[3]Graduate School of Information Science and Technology, Osaka University, Osaka 565-0871, Japan

Corresponding author: Sorachi Kato (kato.sorachi@ist.osaka-u.ac.jp)

ABSTRACT This study aims to determine the upper limit of the wireless sensing capability of acquiring physical space information. This is a challenging objective because, at present, wireless sensing studies continue to succeed in acquiring novel phenomena. Thus, although we have still not obtained a complete answer, a step is taken toward it herein. To achieve this, CSI2Image, a novel channel state information (CSI)-to-image conversion method based on generative adversarial networks (GANs), is proposed. The type of physical information acquired using wireless sensing can be estimated by checking whether the reconstructed image captures the desired physical space information. We demonstrate three types of learning methods: generator-only learning, GAN-only learning, and hybrid learning. Evaluating the performance of CSI2Image is difficult because both the clarity of the image and the presence of the desired physical space information must be evaluated. To solve this problem, we propose a quantitative evaluation methodology using an image-based object detection system. CSI2Image was implemented using IEEE 802.11ac compressed CSI, and the evaluation results show that CSI2Image successfully reconstructs images. The results demonstrate that generator-only learning is sufficient for simple wireless sensing problems; however, in complex wireless sensing problems, GANs are essential for reconstructing generalized images with more accurate physical space information.

INDEX TERMS Wireless sensing, channel state information, deep learning, generative adversarial networks, image reconstruction.

I. INTRODUCTION

This study considers the upper limit of the wireless sensing capability of acquiring physical space information. Wireless sensing enables us to obtain various physical space data by deploying access points (APs). Several studies have already shown the possibility of extracting physical space information from radio waves. In particular, channel state information (CSI)-based methods are improving the practical feasibility of wireless sensing because CSI, which is used for multiple-input multiple-output (MIMO) communication,

is easily acquired from commercial Wi-Fi devices. Using Wi-Fi CSI, state-of-the-art studies have already achieved remarkable results. In the future, Wi-Fi may become a sensing platform; the IEEE 802.11 wireless LAN working group has established a study group for WLAN sensing. The details of wireless sensing are discussed in Section II-A.

This study attempts to reconstruct images from CSI obtained from off-the-shelf Wi-Fi devices to understand the upper limit of the wireless sensing capability of acquiring physical space information. If the conversion from CSI to images corresponding to the physical space can be realized, the possibility of extracting physical space information using CSI can be approximately estimated. In addition, because the

The associate editor coordinating the review of this manuscript and approving it for publication was P. Venkata Krishna.
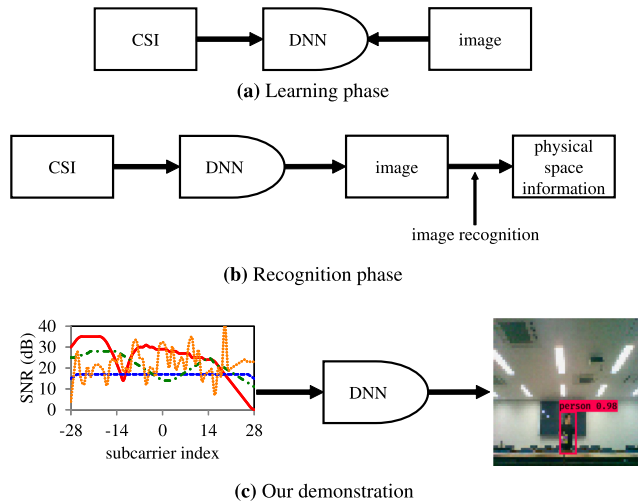
FIGURE 1. Application example of CSI2Image with object detection.

eye is the most high-resolution sensor in the human body, the images serve as human-understandable information. Furthermore, object detection technology, which has developed in conjunction with the emergence of deep learning and the next generation of applications such as automated driving, can be used to automatically build learning data without manual labeling.

Figures 1 shows an application example of CSI-to-image conversion: automatic wireless sensing model generation. The generation consists of two phases: the learning phase and the recognition phase. In the learning phase shown in Figure 1(a), the system simultaneously captures the CSI and images of the target space, following which the system trains a deep neural network (DNN) with the captured CSI and images. Finally, the system extracts the physical space information from the image reconstructed from the captured CSI using the trained DNN, as shown in Figure 1(b). Figure 1(c) shows a practical example of automatic wireless sensing model generation; this is demonstrated as an evaluation in Section IV.

This paper proposes CSI2Image, a novel wireless sensing method to convert radio information into images corresponding to the target space using a DNN. To the best of our knowledge, this is the first time CSI-to-image conversion has been achieved using GANs. From the perspective of CSI-to-image conversion without GANs, a few related studies have been conducted [1], [2]. Wi2Vi [1] uses the video recovered by CSI when a security camera is disabled owing to power failure, malfunction, or attack. Under normal conditions, Wi2Vi extracts the background image from the camera image, detects a person using the difference between the background image and the image, and learns by associating it with the CSI. Under critical conditions, Wi2Vi generates an image by superimposing the detected user onto the background image. [2] has successfully generated pose images generated from skeletons from CSI by learning the relationship between the skeleton model of human posture and CSI.

References [1], [2] are application-specific approaches, using application-specific information such as background images and skeleton models. By contrast, the present study focuses on a general-purpose CSI-to-image conversion method using GANs.

The main contributions of this paper are as follows:

- The use of GANs for CSI-to-image conversion was proposed, implemented, and evaluated. In particular, because simply introducing GANs is insufficient, this paper shows three methods of learning the conversion model: generator-only learning, GAN-only learning, and hybrid learning.
- Two applications, material sensing and device-free user localization, are developed and evaluated to show the versatility of CSI2Image. CSI2Image enables us to easily develop wireless sensing applications.
- Novel position-detection-based quantitative evaluation methodology with you only look once (YOLO), which is an image-based object detection system, to evaluate the performance of CSI-to-image conversion is demonstrated. Specifically, Section IV quantitatively shows that the use of GANs enables the successful reconstruction of more generalized images from CSI compared to generator-only learning.
- Empirical evaluation using off-the-shelf devices is performed using compressed CSI, which can be acquired from IEEE 802.11ac devices. The obtained results can be easily reproduced using an off-the-shelf USB camera, a Raspberry Pi, and a packet capture tool.

The remainder of this paper is organized as follows. Section II describes related works on wireless sensing and GANs. Section III proposes CSI2Image with three generator learning structures: generator-only learning, GAN-only learning, and hybrid learning. Section IV presents the qualitative and quantitative evaluation of the three learning structures proposed in Section III. The quantitative evaluation methodology is also a proposal for the evaluation of CSI-to-image conversion. Finally, the conclusions are presented in Section V.

II. RELATED WORKS
The present work explores the areas of wireless sensing and GANs.

A. WIRELESS SENSING
Several studies have already shown the possibility of extracting physical space information from radio waves using wireless sensing. This has been applied for various purposes, including device localization [3]–[7], [22]–[29], device-free user localization [8], [9], [23], [30]–[32], gesture recognition [10], [11], [33], [34], device-free motion tracking [12], [35], [36], RF imaging [37]–[39], crowdedness estimation [40], activity recognition [13], [14], respiratory monitoring [15], [41], heart rate monitoring [41], material sensing [16], [42], soil sensing [17], keystroke recognition [18], emotion recognition [19], in-body device

TABLE 1. Summary of wireless sensing using CSI.

Method	Context	Performance
SpotFi [3]	Device Localization	Median localization error of $40\,\text{cm}$
Chronos [4]	Device Localization	Median localization error of $65\,\text{cm}$ and $98\,\text{cm}$ for LOS and NLOS scenarios, respectively,
UAT [5]	Device Localization	Localization error under $1\,\text{m}$ at $90\,\%$ of predefined locations
SparseLoc [6]	Device Localization	Approximately $80\,\%$ localization errors are smaller than $2\,\text{m}$
DS-3DCNN [7]	Device Localization	Mean localization error of $0.984\,\text{m}$
LiFS [8]	Device-free user localization	Median accuracy of $0.5\,\text{m}$ and $1.1\,\text{m}$ in LOS and NLOS scenarios, respectively,
Fukushima et al. [9]	Device-free User Localization	$96\,\%$ Accuracy
CrossSense [10]	gesture recognition	average accuracy of $94.5\,\%$ and $98.5\,\%$ for CSI-based gait identification and gesture recognition, respectively,
WiGrus [11]	Gesture recognition	Average accuracy of $96\,\%$ and $92\,\%$ in the LOS and NLOS scenarios, respectively,
WiDraw [12]	Handwriting recognition, Hand tracking	Track the user's hand with a median error lower than $5\,\text{cm}$, recognize handwriting words with an accuracy of $91\,\%$
CARM [13]	Human activity recognition	Average accuracy of over $96\,\%$
EI [14]	Human activity recognition	Classification accuracy of up to $75\,\%$
Hillyard et al. [15]	Respiratory Monitoring	Median error rate of $0.60\,\text{bpm}$
IntuWition [16]	Material sensing	Average accuracy of $95\,\%$ and $92\,\%$ for LOS and NLOS situations, respectively, in classifying five types of materials
Strobe [17]	Soil sensing	Estimates moisture at saturation with errors less than $0.03\,\text{m}^3/\text{m}^3$
WiKey [18]	Keystroke detection	Average accuracy of $97.5\,\%$ and $96.4\,\%$ for detecting the keystroke and classifying single keys, respectively, recognize keystrokes in a continuously typed sentence with an accuracy of $93.5\,\%$ in real-world experiments
EQ-Radio [19]	Emotion Recognition	Average accuracy of $87\,\%$
Ohara et al. [20]	Object state change detection	Average accuracy of approximately $80\,\%$ for detecting state change in furniture
WiCapture [21]	Position tracking	Median accuracy of $0.88\,\text{cm}$ compared to infrared-based tracking systems

localization [43], object state change detection [20], touch sensing [44], device proximity detection [45], device orientation tracking [46], and human detection through walls [38], [47], [48]. While these studies have explored new possibilities using an application-specific approach, the present work is unique in that it attempts to construct a general-purpose wireless sensing technique.

In terms of the physical layer, the proof of concept has been demonstrated in wireless communication devices such as specially customized hardware [19], [30], [33], [38], [41], [43], [44], [47], [49], mmWave [29], [36], UWB [37], [39], RFID [23], [24], [26], [46], [48], [50]–[52], LoRa [53], IEEE 802.15.4 [40], Bluetooth [27], [54], IEEE 802.11n [3], [7], [8], [10], [13], [14], [16]–[18], [21], [31], and IEEE 802.11ac [9], [55]. The use of special customized hardware such as USRP [11], [56] and WARP [57] enables the extraction of more detailed physical space information. However, the use of commercially available equipment such as IEEE 802.11n, RFID, Bluetooth, and IEEE 802.15.4 is advantageous for deployment and the reproducibility of research results. In particular, the emergence of CSI tools [58]–[61] has been particularly significant for the wireless sensing research community. Commercially available IEEE 802.11n devices have been used not only to produce different research
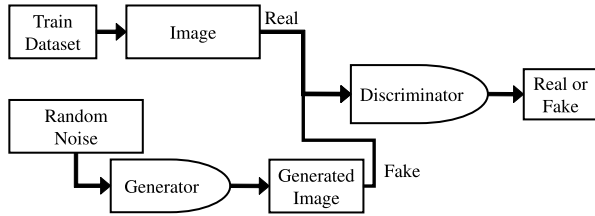
FIGURE 2. DCGAN.

results, but have also opened up possibilities for the deployment of wireless sensing. However, at present, research using IEEE 802.11n faces the problem that only one section of IEEE 802.11n devices, Intel 5300 NIC, Atheros AR9390, AR9580, AR9590, AR9344, or QCA9558, can obtain CSI. The present study uses the IEEE 802.11ac [62], [63] compressed CSI. The IEEE 802.11ac compressed CSI is standardized to reduce the overhead of CSI feedback. Compressed CSI can be acquired from any device that supports IEEE 802.11ac or IEEE 802.11ax.

B. GENERATIVE ADVERSARIAL NETWORKS

GANs enable the generation of new data with the same statistics as the training data using a generative model [64], and they have been used in several applications [65], [66]. The generative model is constructed by alternately learning a generator and a discriminator to trick the discriminator. This section introduces deep convolutional GAN (DCGAN) [67] and super-resolution GAN (SRGAN) [68], both of which are highly relevant to this study.

DCGAN constructs a generative model to generate realistic fake images from random noise [67]. Figure 2 shows the model structure of DCGAN. The DCGAN trains the discriminator to identify an image as real when the image is from the training dataset and as fake when generated from random noise by the generator. Simultaneously, the DCGAN trains the generator to generate images (from random noise) that the discriminator identifies as real. The generator is implemented using deep convolutional neural networks [69]. As the generator and the discriminator learn to compete with each other, the generator generates high-quality fake images.

SRGAN generates high-resolution images from corresponding low-resolution images [68]. Figure 3 shows the model structure of SRGAN. SRGAN trains the discriminator to identify an image as real when it is from the training dataset and as fake when generated from a low-resolution image by the generator. Simultaneously, SRGAN trains the generator to generate images (from low-resolution images) that will be identified as real by the discriminator. From DCGAN and SRGAN, it can be said that GANs can be used to create fake data that appear real or recover real data from small amounts of data.

III. CSI2Image: IMAGE RECONSTRUCTION FROM CSI

Figure 4 shows the entire system of the proposed CSI2Image. CSI2Image is composed of training data, a generator, and
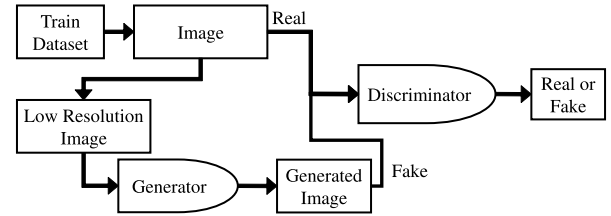

FIGURE 3. SRGAN.
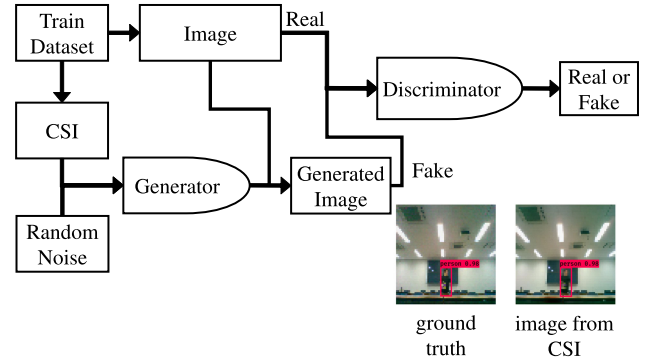

ground   image from
truth   CSI

FIGURE 4. Overview.

a discriminator. Section III-A shows the details of the training data, Section III-B shows the model structure of the generator, and Section III-C shows the model structure of the discriminator. This paper proposes three types of generator learning methods: generator-only learning, GAN-only learning, and hybrid learning. Generator-only learning does not use a discriminator, as described in Section III-D.

A. TRAINING DATA

The training data of CSI2Image consist of simultaneously captured images and CSIs. Full-color 64 × 64 pixel images and compressed CSI are used. Compressed CSI is used in off-the-shelf APs, smartphones, and PCs for their wireless communications, and the standard format of CSI feedback is as specified in the IEEE 802.11ac standard [62], [63].

CSI2Image recovers the right singular matrix V from compressed CSI and uses the first column of V as input data. Note that the singular value decomposition of the CSI is expressed as follows.

$$\mathbf{CSI} = USV^H$$

where U is a left singular matrix, S is a diagonal matrix with singular values of CSI, and V is a right singular matrix.

The compressed CSI in IEEE 802.11ac includes the angle information ϕ and ψ. V is calculated with ϕ and ψ by Equation (1).

$$V = \left\{ \prod_{k=1}^{min(N,M-1)} \left[D_k \prod_{l=k+1}^{M} G_{l,k}^T(\psi_{k,k}) \right] \right\} \widetilde{I}_{M \times N}. \quad (1)$$

where M is the number of RX antennas, N is the number of TX antennas, and $\widetilde{I}_{M \times N}$ is the identity matrix in which zeros
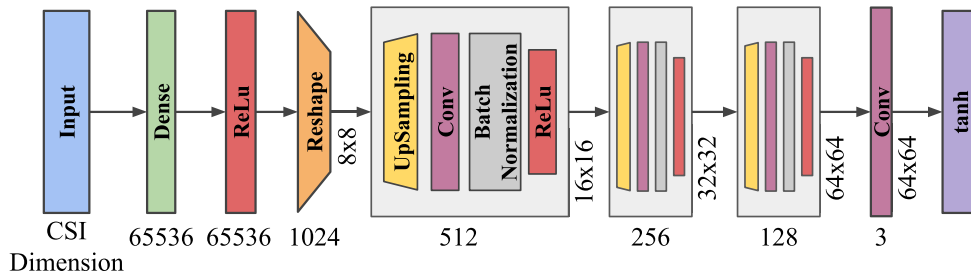
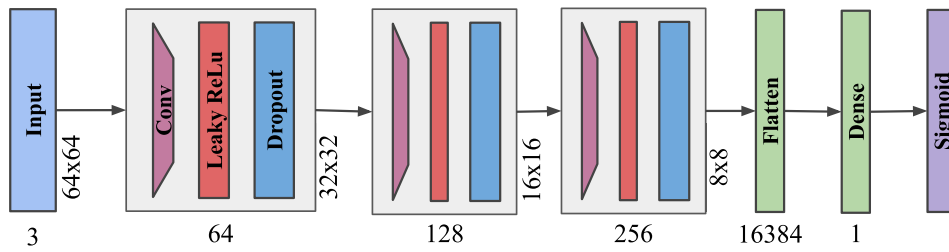**FIGURE 5.** Generator network.

FIGURE 6. Discriminator network.

are inserted in the missing element if $N \neq M$. D_k is a diagonal matrix that is expressed as follows:

$$
D_k = \begin{pmatrix}
I_{k-1} & 0 & 0 & \cdots & 0 \\
0 & e^{j\phi_{k,k}} & 0 & \cdots & 0 \\
0 & 0 & \ddots & 0 & 0 \\
\vdots & \vdots & 0 & e^{j\phi_{M-1,k}} & 0 \\
0 & 0 & 0 & 0 & 1
\end{pmatrix}
$$

where I_{k-1} is $(k-1) \times (k-1)$ identity matrix. $G_{l,k}(\psi)$ is the Givens rotation matrix

$$
G_{l,k} = \begin{pmatrix}
I_{k-1} & 0 & 0 & 0 & 0 \\
0 & cos(\psi) & 0 & sin(\psi) & 0 \\
0 & 0 & I_{l-k-1} & 0 & 0 \\
0 & -sin(\psi) & 0 & cos(\psi) & 0 \\
0 & 0 & 0 & 0 & I_{M-1}
\end{pmatrix}
$$

where I_{k-1} is $(k-1) \times (k-1)$ identity matrix.

B. GENERATOR NETWORK

Figure 5 shows the network structure of the generator. The compressed CSI is input to the dense layer of 65,536 neurons with the rectified linear unit (ReLU) layer, and the neurons are reshaped into an $8 \times 8 \times 1024$ tensor. The dimension of compressed CSI depends on the channel width. The tensor enters the upsample layer, convolution layer with 3×3 kernel, batch normalization layer, and ReLU layer three times. Finally, it is also input to the convolution layer with the 3×3 kernel and activation function of tanh to obtain an output of $64 \times 64 \times 3$ tensor. Adam is utilized as the optimizer of the generator network, whose learning rate is 0.0002, and the momentum term is 0.5. The loss function for the generator network is the mean squared error (MSE) [70].

TABLE 2. Layer parameter details.

Upsampling Layer	size = 2×2
Convolution Layer	Kernel size = 3×3, stride = 2
BatchNormalization Layer	momentum = 0.8
LeakyReLU Layer	alpha = 0.2
Dropout Layer	rate = 0.25

C. DISCRIMINATOR NETWORK

Figure 6 shows the network structure of the discriminator. The input is a full-color image of 64×64 pixels. The color image is then fed into four sets of the convolution layer of a 3×3-size kernel with stride 2, batch normalization, LeakyReLU function ($\alpha = 0.2$), and a dropout of 0.25. The output is then flattened and activated by a sigmoid function. The output value is the range of 0 to 1. The discriminator network uses the Adam optimizer, whose initial setting is the same as that of the generator network, and the loss function of binary cross-entropy.

D. LEARNING PHASE

In this work, three methods are proposed for the learning phase: generator-only learning, GAN-only learning, and hybrid learning. Generator-only learning learns the correlation between compressed CSIs and images. GAN-only learning uses both a generator and a discriminator. Hybrid learning combines generator-only and GAN-only learning.

E. GENERATOR-ONLY LEARNING

Figure 7 and Algorithm 1 depict the model structure and pseudo-code, respectively, of the generator-only learning. The convolutional-neural-network-based generator is trained with the measured CSIs and simultaneously captured images.
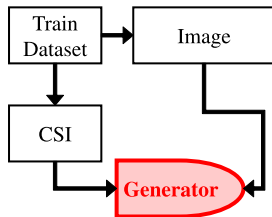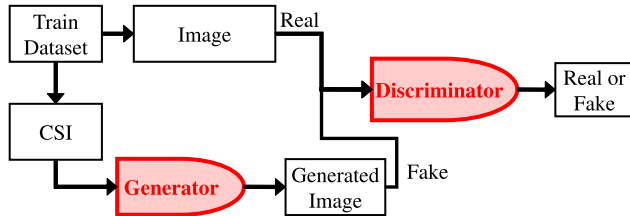
FIGURE 7. Generator-only learning.

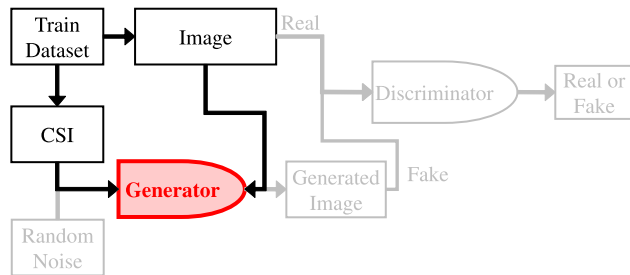**FIGURE 8.** GAN-only learning.

FIGURE 9. Hybrid learning: CSI2Image learning.

As generator-only learning learns the relations between the CSIs and images given as the training data, the generator may not accurately generate images from unknown CSIs.

Algorithm 1 Generator-Only Learning

1: $N \Leftarrow$ number of training iterations
2: **for** $i = 1$ to N **do**
3:     csi_list $\Leftarrow$ batch of CSI
4:     real_images $\Leftarrow$ batch of real images
5:     G.csi2image.train(csi_list, real_images)
6: **end for**

F. GAN-ONLY LEARNING

Figure 8 and Algorithm 2 show the model structure and pseudo-code, respectively, of GAN-only learning. Because the discriminator learns the converted image while judging whether it is a real image, this method is more likely to reconstruct a clear image than generator-only learning. However, the discriminator only judges the legitimacy of the converted image, and it may not convert an image corresponding to the measured and compressed CSI. In particular, the discriminator may not learn the detailed parts of the image.

G. HYBRID LEARNING

Hybrid learning is the integration of generator-only learning and GAN-only learning. The following four steps are regarded as one training epoch:
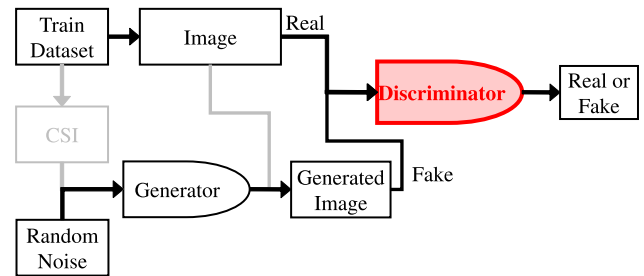

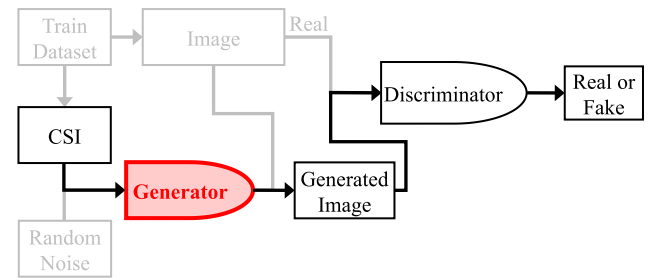**FIGURE 10.** Hybrid learning: discriminator learning.

FIGURE 11. Hybrid learning: generality learning.

Algorithm 2 GAN-Only Learning

1: $N \Leftarrow$ number of training iterations
2: **for** $i = 1$ to N **do**
3:     csi_list $\Leftarrow$ batch of CSI
4:     real_images $\Leftarrow$ batch of real images
5:     D.train(real_images, REAL)
6:     D.train(G.generate(csi_list), FAKE)
7:     G.generality.train(csi_list, REAL)
8: **end for**

---

1) CSI2Image learning (Figure 9)
2) Image reconstruction via the generator
3) Discriminator learning (Figure 10)
4) Generality learning (Figure 11)

Algorithm 3 describes the pseudo-code of hybrid learning. Let N denote the number of training iterations and K denote the interval for performing generality learning. Lines 4 to 6 of Algorithm 3 represent CSI2Image learning. CSI2Image learning obtains the training data and then trains the generator using the measured and compressed CSIs and the images corresponding to the CSIs at line 6, as shown in Figure 9. Lines 7 to 8 of Algorithm 3 represent discriminator learning. At line 7, the discriminator is trained to assess the training image to be real, and at line 8, it is trained to assess the generated image (obtained from the random noise) to be fake, as shown in Figure 10. Lines 9 to 11 of Algorithm 3 represent generality learning. The generator is trained every K epochs by feeding the compressed CSI to judge the generated image to be real by the discriminator, as shown in Figure 11. When the value of K is large, the generalization performance increases, while the CSI information is lost. When the value of K is small, the image quality reduces because of generality loss.

Algorithm 3 Hybrid Learning

1: $N \Leftarrow$ number of training iterations
2: $K \Leftarrow$ interval of generality training
3: **for** $i = 1$ to N **do**
4:    csi_list $\Leftarrow$ batch of CSI
5:    real_images $\Leftarrow$ batch of real images
6:    G.csi2image.train(csi_list, real_images)
7:    D.train(real_images, REAL)
8:    D.train(G.generate(random_noise), FAKE)
9:    **if** i mod $K == 0$ **then**
10:      G.generality.train(csi_list, REAL)
11:   **end if**
12: **end for**

H. IMAGE GENERATION PHASE

In the image generation phase, the compressed CSI measured by wireless devices is fed into the pre-trained generator, and the generator converts the CSIs into full 64×64 pixel images, as shown in Figure 12.
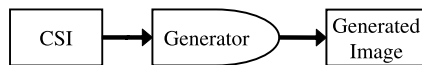
FIGURE 12. Image generation phase.

IV. EVALUATION

We evaluated the CSI2Image using material sensing and device-free user localization tasks as examples of wireless sensing applications.

A. EXPERIMENT 1: MATERIAL SENSING

1) EVALUATION SETTINGS

To clarify whether CSI2Image is suitable for a material sensing system, we evaluated the CSI2Image using three different daily containers: a milk package (Milk), a plastic bottle (Bottle), and a drink can (Can), as shown in Figure 15(a), (b), and (c). A milk package is made of paper, 1000 ml volume, rectangular, and has milk inside. A plastic bottle is made from PET, 500 ml volume, cylindrical, and has water inside. A drink can is made of aluminum, 355 ml volume, cylindrical, and has carbonated drink inside. Figure 13 shows the configuration of each piece of equipment in the material sensing experiments. Figure 14 demonstrates a snapshot of the environment. These three containers are located between an access point and a device. No other moving objects existed in the same room, and the orientation and container positions did not change during the experiments. This experiment utilized an AP, a camera, a computer, and a capture device as a compressed CSI sniffer. In material sensing experiments, the AP was a TP-Link Archer C6 with 2 TX antennas, the camera was a Panasonic CF-SX1GEPDR with a resolution of 1280×720 pixels, the computer was a MacBook Pro (13-inch, 2020) with 2 RX antennas, and the capture device was a Raspberry Pi 3B+ with Raspbian 5.4.79.
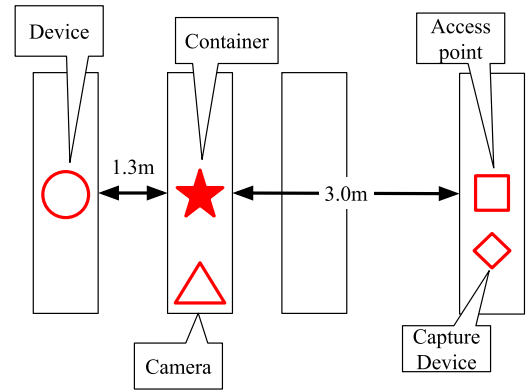
FIGURE 13. Experimental environment for material sensing.

FIGURE 14. Snapshot of the experimental environment.

The channel width was 80 MHz including 234 subcarriers. The evaluation used 3506 images as training data and 500 images as test data. The model for the CSI-to-image conversion was made using the proposed hybrid learning method. The model was trained using an Amazon EC2 instance equipped with an Intel Xeon CPU E5-2686 central processing unit running at a clock speed of 2.3 GHz on eighteen cores, 64 GB of DDR-SDRAM, a Tesla K80 graphics processing unit running at a clock speed of 560 MHz on 4992 cores, and a solid-state drive for storage. The number of epochs was 15,000, and the batch size was 16. For this hybrid learning, the parameter K was set to 8.

2) EVALUATION RESULTS

Figure 15 shows the examples of the generated image. CSI2Image succeeded in generating images of three containers. Figure 16 shows the confusion matrix for three containers' generated images. CSI2Image generates correct images with 97 %, 96 %, and 90 % accuracy for a milk package, plastic bottle, and drink can, respectively. CSI2Image achieves high detection accuracy for both the milk package and the plastic bottle. There are cases where a drink can is recognized as a plastic bottle, although the rate was low at approximately 5 %. We think this is because a plastic bottle and a drink can be both cylindrical.
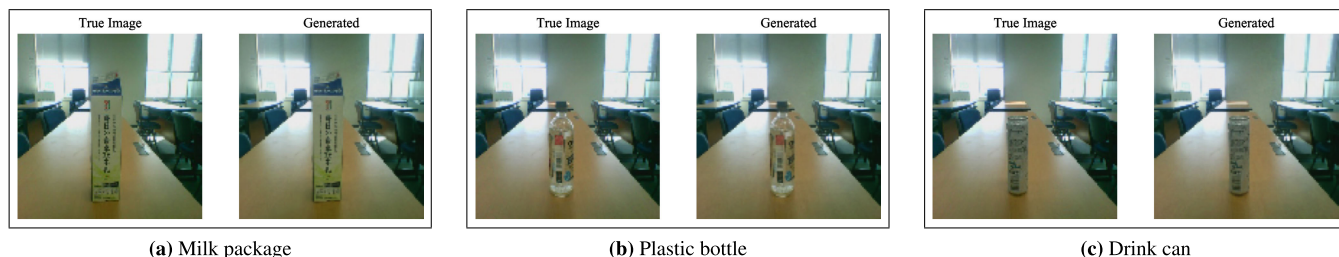
(a) Milk package   **(b)** Plastic bottle   **(c)** Drink can

FIGURE 15. Experiment 1: Examples of successful material sensing.

FIGURE 16. Experiment 1: Confusion matrix for three objects image reproduction.

FIGURE 17. Experimental environment for device-free localization.

B. EXPERIMENT 2: DEVICE-FREE USER LOCALIZATION

As shown in Section IV-A, we found that CSI2Image can be used for material sensing. To demonstrate the versatility of CSI2Image and evaluate its properties more quantitatively, we evaluated it in device-free user localization by gradually increasing the complexity of the problem. Because the conversion of CSI to images is a new research area, no quantitative evaluation methodology has been established. Therefore, we propose a novel quantitative evaluation methodology using device-free user localization to combine YOLO, an image-based object detection system.

1) EVALUATION SETTINGS

Figure 17 shows the configuration of each piece of equipment in the device-free localization experiments. Figure 14 demonstrates a snapshot of the environment. In device-free localization experiments, the AP was a Panasonic EA-7HW04AP1ES with 3 TX antennas. The camera was a Panasonic CF-SX1GEPDR with a resolution of 1280×720 pixels, the computer was a MacBook Pro (13-inch, 2017) with 2 RX antennas, and the capture device was a Panasonic CF-B11QWHBR with CentOS 7.7-1908. The channel width was 20 MHz including 52 subcarriers. The proposed CSI2Image model was developed using a Dell Alienware 13 R3 computer equipped with an Intel Core i7-7700HQ central processing unit running at a clock speed of 3.8 GHz on four cores, 16 GB
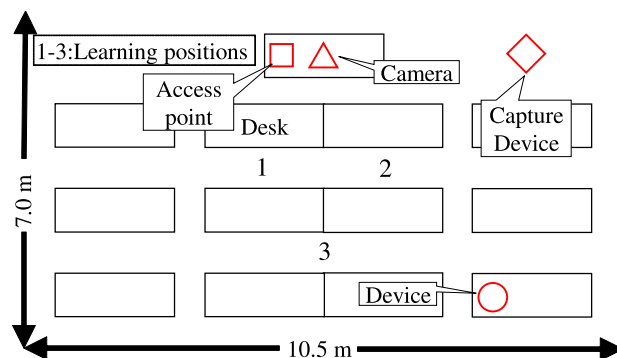
of DDR-SDRAM, a Geforce GTX 1060 Mobile graphics processing unit running at a clock speed of 1404 MHz on 1280 cores, and a solid-state drive for storage. The object detection library used was YOLO v3 [71], [72], and the model data were trained with [73] from the COCO dataset [74]. The threshold to determine the object detection in YOLO is 0.3.

2) SINGLE USER

In this evaluation, three methods were compared:
- generator-only learning (gonly),
- GAN-only learning (gan), and
- hybrid learning (hybrid),

which are described in Section III-D. The following four aspects were extracted to evaluate these methods quantitatively:

1) Object detection success rate. The high score obtained indicates that the quality of the generated images is sufficient for object detection.
2) Average confidence score when object detection is successful. The confidence score is the confidence level of the object recognition algorithm in outputting the recognition result.
3) Structural similarity (SSIM) [75]. SSIM is a standard measure of image quality used to evaluate the performance of image processing algorithms, such as image compression.
4) Position detection accuracy rate. The position detection accuracy rate is the percentage of correct locations detected via object recognition.
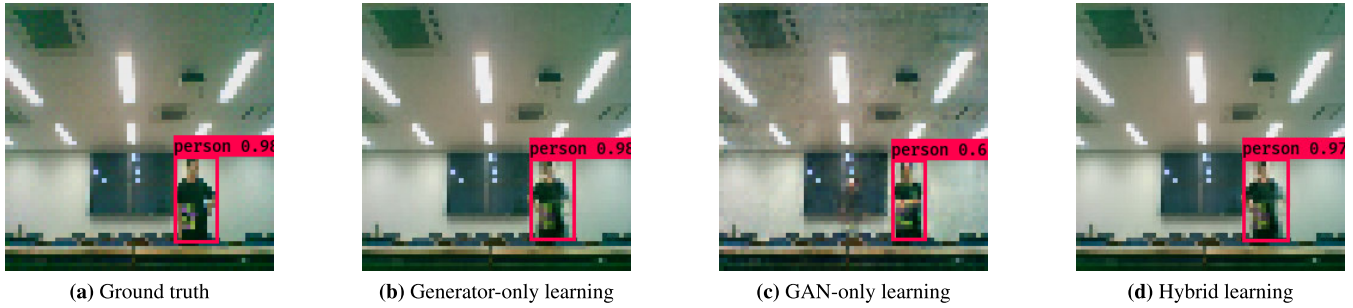
FIGURE 18. Experiment 2: Examples of successful position detection with one user.
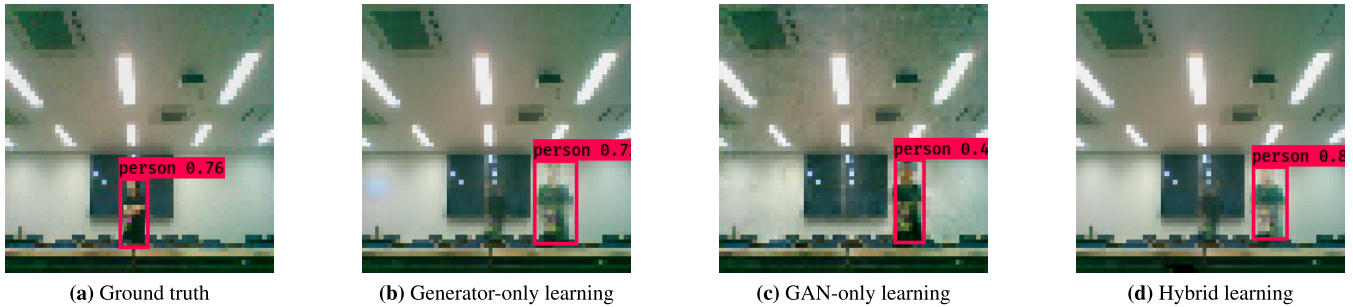

FIGURE 19. Experiment 2: Examples of failed position detection with one user.

The SSIM index is expressed by Equation 2,

$$SSIM = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (2)$$

where x and y are vectors, whose elements are the pixels of the original image and the reconstructed images, respectively. Let μ_x and μ_y denote the average pixel values of images x and y, σ_x, and σ_y be the standard deviations of images x and y, and σ_{xy} be the covariance of images x and y, respectively. Both C_1 and C_2 are constant values defined as $C_1 = (255 \, K_1)^2$ and $C_2 = (255 \, K_2)^2$, respectively. In this case, the parameters of $K_1 = 0.01$ and $K_2 = 0.03$ are the same as those in [75]. The SSIM index takes a value from 0 to 1, where 1 represents an exact image match.

The experiment was performed with only one person at positions 1 to 3 in Figure 17. Three types of image patterns were possible, in which the person would be at positions 1, 2, or 3, respectively. The evaluation used 180 images as training data and 184 images as test data. The number of epochs was 32,000, and the batch size was 32. In hybrid learning, K was 8.

Figure 18 shows an example of successful position detection with one user. The red square on each figure represents the object detection results obtained using YOLO. If a person is detected on the right of the image, as shown in Figure 18(a), the position detection is accurate. The positions of generator-only and hybrid learning in Figures 18(b) and 18(c) are accurate, as is the shape of the person. On the other hand, GAN-only learning, shown in Figure 18(d), accurately detects the position of the person, while the shadow of the person is also output in the center of the incorrect position.

Figure 19 shows an example of failed position detection with one user. If a person is detected on the right of the image, as shown in Figure 19(a), the position detection is accurate. As can be seen from Figures 19(b) to 19(c), pale ghost-like shadows appear at the middle and the right of the images. In contrast, GAN-only learning in Figure 19(d) shows that the GAN produces a clean image as compared to generator-only and hybrid learning, although the position is inaccurate. We assume that the GAN-only learning achieves clean image generation because the GAN successfully learns and generates human images owing to its generalization performance. On the other hand, because of the high generalization performance, the correlation between the CSI and the person's position weakens, and as a result, there are cases where the person is displayed in the wrong position.

Figure 20(a) shows the success rate of human detection. The black and white bars represent the results using the training and test data, respectively. The confidence threshold of YOLO is 0.3. GAN-only learning achieved the highest score in terms of the detection success rate. In the test data, the detection success rate of generator-only learning, GAN-only learning, and hybrid learning were approximately 92.7 %, 93.5 %, and 92.3 %, respectively.

Figure 20(b) shows the average confidence score when the object detection is successful. In addition to the detection success rate described above, GAN-only learning achieved the highest score. In the test data, the average confidence scores of generator-only learning, GAN-only learning, and hybrid learning were approximately 91.1 %, 94.5 %, and 90.8 %, respectively. Among the three methods, GAN-only learning achieved the highest score for both the successful detection rate and confidence score of YOLO, especially
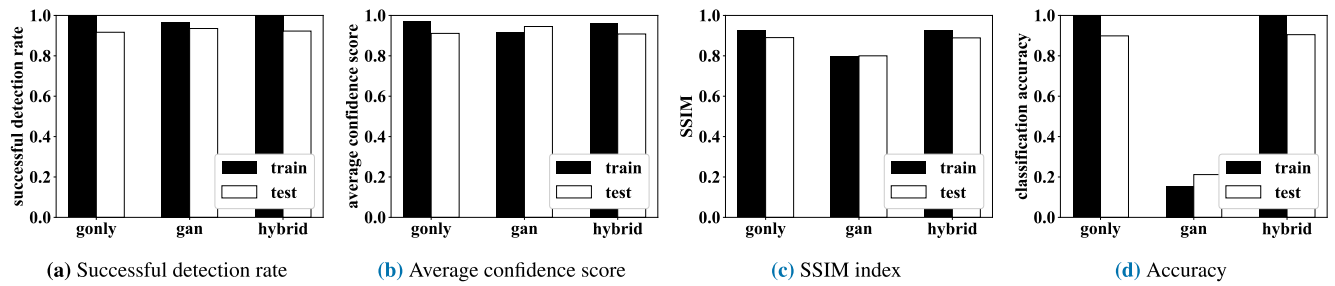
(a) Successful detection rate **(b)** Average confidence score **(c)** SSIM index **(d)** Accuracy

FIGURE 20. Experiment 2: Quantitative evaluation of single-user position classification.

(a) Ground truth **(b)** Generator-only learning **(c)** GAN-only learning **(d)** Hybrid learning
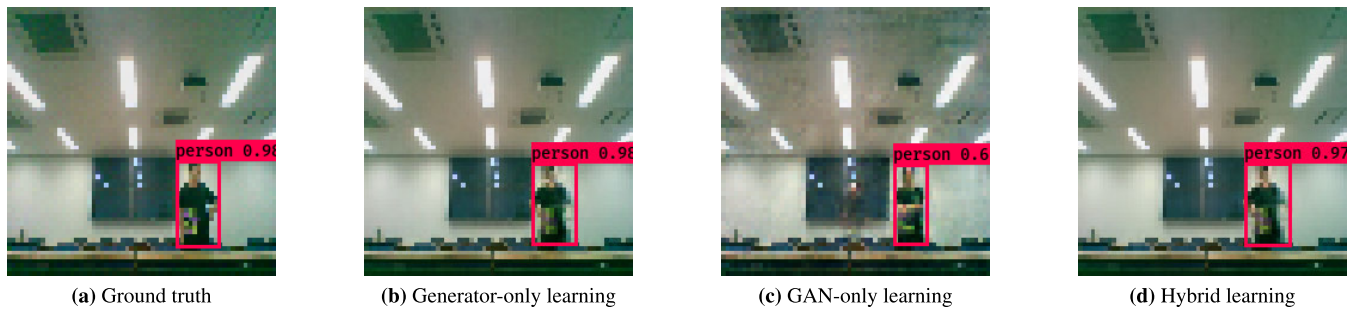
FIGURE 21. Experiment 2: Examples of successful position detection with one or two users.

for the confidence score, the GAN succeeded in generating images that achieved a higher confidence score than the training data. We assume this is because the GAN is specialized to generate realistic images, although the positioning accuracy of GAN-only learning is relatively low because of its high generalization performance, as described above.

Figure 20(c) shows the SSIM index of each comparison method. In contrast to the detection success rates and the average confidence score, GAN-only learning showed the worst performance in terms of the SSIM index. With the test data, the SSIM index results were approximately 0.890 for generator-only learning, 0.800 for GAN-only learning, and 0.889 for hybrid learning.

To understand the low SSIM performance of GAN-only learning, we evaluated the position detection accuracy. The results show that GAN-only learning had the worst performance compared to generator-only and hybrid learning. With the test data, the accuracies were found to be approximately 89.9 % for generator-only learning, 21.2 % for GAN-only learning, and 90.5 % for hybrid learning.

Thus, although the detection success rate and the average confidence score are the highest in GAN-only learning, the SSIM index is low owing to misplaced-user images. In particular, GAN-only learning has position detection accuracy even with the training data. This is because GAN-only learning only learns the legitimacy of the generated image using the discriminator, as shown in Figure 20(d).

3) ONE OR TWO USERS
For more complex situations, the position detection was evaluated for the case of one or two users. Specifically, six

types of classification problems were evaluated when one person or two people were at positions 1 to 3, as shown in Figure 17: "one person at 1," "one person at 2," "one person at 3," "two people at 1 and 2," "two people at 1 and 3," and "two people at 2 and 3." We used 720 images as training data and 330 images as test data for the evaluation. The other conditions were identical to those presented in Section IV-B2.

Figure 21 shows an example of successful position detection with one or two users. If a person can be detected at the center of the image, as shown in Figure 21(a), the position detection is accurate. The positions obtained via generator-only and hybrid learning in Figures 21(b) and 21(d) are accurate, and the human shape is clearly displayed. However, as shown in Figure 21(c), GAN-only learning accurately detects the position, but a shadow is also output on the left side of the incorrect position. As described in the evaluation of single-user localization, GAN-only learning trains the model to be generalized as to the position of a user; thus, we assume that even if a user is displayed at the correct place, the generalized model tends to generate an extra image of a user at another location.

Figure 22 shows an example of failed position detection with one or two users. If a person is detected on the right side of the image, as shown in Figure 22(a), the position detection is accurate. As shown in Figures 22(b) and 22(d), YOLO does not detect the person on the right side of the image for generator-only and hybrid learning, although human-like objects are displayed on the right side. In addition, GAN-only learning shows people in the middle and left of the incorrect position, as shown in Figure 22(c). In this case, we assume that generator-only learning and hybrid learning generate
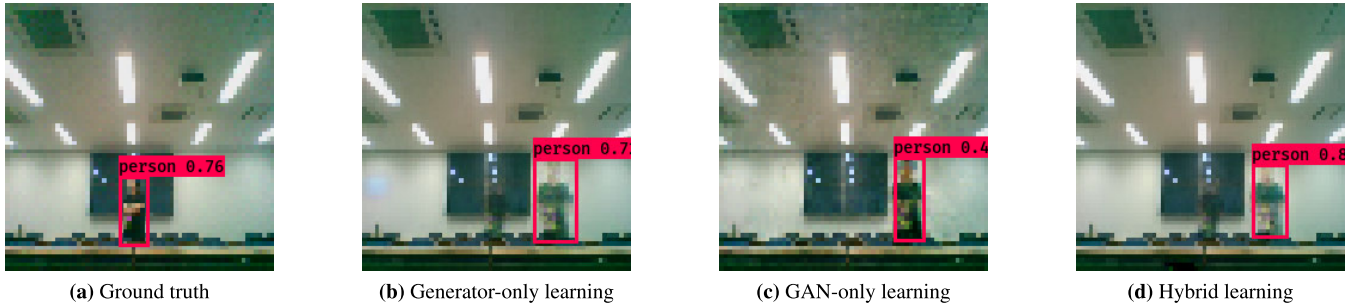
(a) Ground truth	(b) Generator-only learning	(c) GAN-only learning	(d) Hybrid learning

FIGURE 22. Experiment 2: Examples of failed position detection with one or two users.

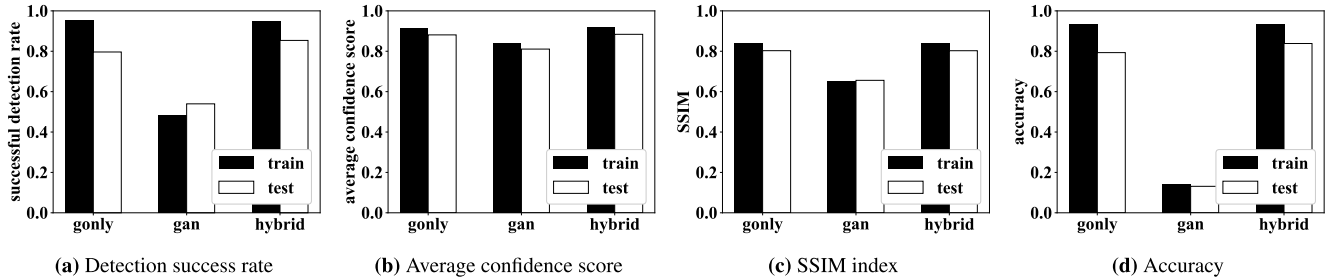| (a) Detection success rate	(b) Average confidence score	(c) SSIM index	(d) Accuracy

FIGURE 23. Experiment 2: Quantitative evaluation of the position classification of one or two users.

fuzzy images because the input CSI contains little information about the position of a user.

Figure 23(a) shows the successful detection rates of each comparison method. It can be observed that hybrid learning achieves the highest detection rate: with the test data, the detection success rates are approximately 79.6 % for generator-only learning, 54.0 % for GAN-only learning, and 85.4 % for hybrid learning. The detection success rate of GAN-only learning is low even when using training data.

Figure 23(b) shows the average confidence score of each comparison method. Similar to the above successful detection rate, it can be observed that the confidence score of hybrid learning is the highest, with the test data, the average confidence scores are approximately 88.1 % for generator-only learning, 81.1 % for GAN-only learning, and 88.4 % for hybrid learning.

Figure 23(c) shows the SSIM index of each comparison method. The results are the same as in the single-user evaluation: GAN-only learning shows the worst performance. With the test data, the SSIM indexes were 0.803 for generator-only learning, 0.656 for GAN-only learning, and 0.803 for hybrid learning.

Figure 23(d) shows the position detection accuracy of each comparison method. The results show that using the test data, hybrid learning achieved the highest accuracy, while GAN-only learning had the lowest accuracy: 79.3 % for generator-only learning, 13.1 % for GAN-only learning, and 83.8 % for hybrid learning.

4) CONTINUOUS DEVICE-FREE LOCALIZATION FOR A SINGLE USER

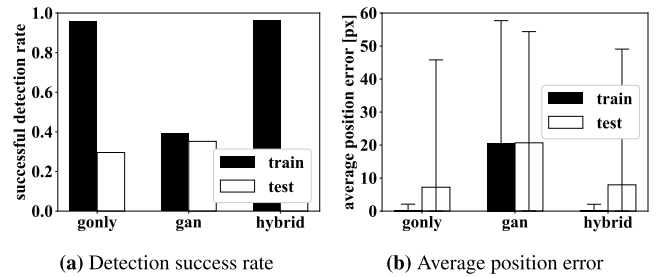To evaluate a more complicated situation than that in Section IV-B3, we conducted experiments in which one

(a) Detection success rate	(b) Average position error

FIGURE 24. Experiment 2: Single-user continuous position estimation.

person walked around an oval connecting positions 1 to 3, as shown in Figure 17. The evaluation used 515 images as training data and 498 images as test data. The other settings were identical to those in Experiment 1 and 2. As the qualitative evaluation results did not differ from those of the position detection problem in Section IV-B2, only the quantitative evaluations are presented in this section.

Figure 24(a) shows the detection success rates of each comparison method. The detection success rates are relatively low compared to those obtained in the evaluations in Sections IV-B2 and IV-B3. This was true even though the training results showed high detection success rates. In the training data, the values were 95.7 % for generator-only learning, 39.0 % for GAN-only learning, and 96.3 % for hybrid learning. In contrast, the test data success rates were 29.6 % for generator-only learning, 35.2 % for GAN-only learning, and 27.8 % for hybrid learning. We believe that the amount of training data is small relative to the complexity of the problem.

Figure 24(a) shows the distance (in pixels) between the left coordinates of the detected box of the training data and
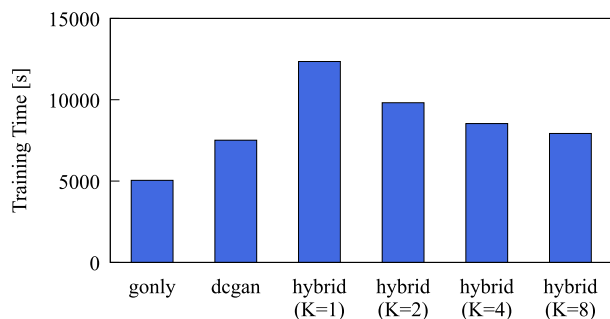
FIGURE 25. Training time in device-free localization evaluation of one user (epoch = 32000).

that of the generated data. The lower value of the distance indicates that the CSI2Image precisely tracks the position of a user. This evaluation only used the generated images that successfully detect a user. The lower limit of the error bar is the minimum value, and the upper limit is the maximum value. The evaluation results show that GAN-only learning cannot be used for single-user continuous position detection. While generator-only learning and hybrid learning are superior to GAN-only learning, they require performance improvements. This is because the maximum value is too high, although the results are only calculated from successfully detected images. Using the test data, the maximum differences of generator-only learning, GAN-only learning, and hybrid learning were 46 px, 54 px, and 49 px, respectively.

C. COMPUTATIONAL COSTS
We now evaluate the computational costs of our CSI2Image with respect to the training times and time for CSI-to-image conversion.

1) COMPUTATIONAL COSTS OF MODEL TRAINING
We compared the training times for the methods of gonly, dcgan, hybrid ($K = 1$), hybrid ($K = 2$), hybrid ($K = 4$), and hybrid ($K = 8$). Training data and the number of training epochs were the same as those of Experiment B-II.

Figure 25 shows the training times for each method: 5045.0 s, 7512.8 s, 12 350.1 s, 9815.8 s, 8535.0 s, 7926.6 s and 7614.1 s for gonly, dcgan, hybrid ($K = 1$), hybrid ($K = 2$), hybrid ($K = 4$), and hybrid ($K = 8$), respectively. Of the three methods, gonly has the shortest time and hybrid has the longest. For the sake of space, we do not go into details, but this trend regarding the training times is almost the same for the other data in our experiments.

2) COMPUTATIONAL COSTS OF CSI-TO-IMAGE CONVERSION
We measure the time taken to convert CSI to images using a pre-trained model. The training data, test data, and pre-trained models are all the same as those used in the material sensing experiments, and the machine on which we performed the conversion was iMac, equipped with an Intel Core i5-8500 central processing unit running at a clock speed of

4.1 GHz on six cores, 16 GB of DDR-SDRAM, Radeon Pro 570X graphics processing unit running at a clock speed of 1105 MHz on 1792 cores and a solid-state drive for storage. We inputted 1169 test data into the trained model and generated images. The time taken to generate all the images was 246.65 s, or 0.21 s per image on average.

V. CONCLUSION
This paper proposed CSI2Image, a GAN-based CSI-to-image conversion method, intending to provide hints for exploring the upper limits of wireless sensing. Specifically, three learning methods have been explored: generator-only learning, GAN-only learning, and hybrid learning. First, we demonstrated that our CSI2Image can be applied to various applications by realizing material sensing and device-free user localization with CSI2Image. For material sensing and device-free user localization of a single user who stays in a particular location, we demonstrated that CSI2Image achieves high accuracy even with a relatively small amount of training data. In addition, for device-free user localization of multiple users who stay in a particular location, we show that hybrid learning combining generator-only learning and GAN is effective. Furthermore, for device-free user localization of a single user who is moving continuously, we found that none of our tested methods (generator-only learning, GAN-only learning, and hybrid learning) achieved high accuracy.

The following are possible future works. The first is to challenge the limit of DNN-based wireless sensing. Using the proposed position-detection-based quantitative evaluation methodology with YOLO, it is possible to evaluate the quality of converting CSI to an image quantitatively and without manual labeling. The evaluation framework enables us to explore DNNs that are more suitable for wireless sensing. Possible methods for improving DNNs include designing new DNNs by visualizing the middle layer to analyze how CSI features are learned, using transfer learning to generate a more general model, and using other DNNs such as recurrent neural networks and LSTMs that address temporal aspects.

The second is to develop learning-less wireless sensing methods from DNNs built using CSI2Image. For example, we can derive equations that directly extract real-space information from CSI by visualizing and analyzing the middle layers of DNNs trained with CSI2Image. The middle layers contain information on the relationship between the subject phenomenon and CSI.

The third is the extension of the input/output to/from CSI2Image. The dimension of the input can be increased by acquiring CSI from multiple devices in the same frequency band or multiple frequency bands such as millimeter-wave or sub-1 GHz, time-series CSI, and so on. The increase of input dimension will improve the performance of CSI2Image. The dimension of the outputs can be increased using a video, which contains time dimensional information, and a point cloud containing three-dimensional space information. The increase of output dimension will increase the information which we can recognize.

REFERENCES

[1] M. H. Kefayati, V. Pourahmadi, and H. Aghaeinia, "Wi2 Vi: Generating video frames from WiFi CSI samples," *IEEE Sensors J.*, vol. 20, no. 19, pp. 11463–11473, Oct. 2020.

[2] L. Guo, Z. Lu, X. Wen, S. Zhou, and Z. Han, "From signal to image: Capturing fine-grained human poses with commodity Wi-Fi," *IEEE Commun. Lett.*, vol. 24, no. 4, pp. 802–806, Apr. 2020.

[3] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "SpotFi: Decimeter level localization using WiFi," in *Proc. Conf. ACM Special Interest Group Data Commun. (ACM SIGCOMM)*, Aug. 2015, pp. 269–282.

[4] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single WiFi access point," in *Proc. 13th USENIX Symp. Netw. Syst. Design Implement. (NSDI)*, Mar. 2016, pp. 165–178.

[5] T.-C. Tai, K. C.-J. Lin, and Y.-C. Tseng, "Toward reliable localization by unequal AoA tracking," in *Proc. 17th Annu. Int. Conf. Mobile Syst., Appl., Services*, Jun. 2019, pp. 444–456.

[6] K. Chen, Y. Mi, Y. Shen, Y. Hong, A. Chen, and M. Lu, "SparseLoc: Indoor localization using sparse representation," *IEEE Access*, vol. 5, pp. 20171–20182, 2017.

[7] Y. Jing, J. Hao, and P. Li, "Learning spatiotemporal features of CSI for indoor localization with dual-stream 3D convolutional neural networks," *IEEE Access*, vol. 7, pp. 147571–147585, 2019.

[8] J. Wang, H. Jiang, J. Xiong, K. Jamieson, X. Chen, D. Fang, and B. Xie, "LiFS: Low human-effort, device-free localization with fine-grained subcarrier information," in *Proc. 22nd Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2016, pp. 243–256.

[9] T. Fukushima, T. Murakami, H. Abeysekera, S. Saruwatari, and T. Watanabe, "Evaluating indoor localization performance on an IEEE 802.11ac explicit-feedback-based CSI learning system," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2019, pp. 1–6.

[10] J. Zhang, Z. Tang, M. Li, D. Fang, P. Nurmi, and Z. Wang, "CrossSense: Towards cross-site and large-scale WiFi sensing," in *Proc. 24th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2018, pp. 305–320.

[11] T. Zhang, T. Song, D. Chen, T. Zhang, and J. Zhuang, "WiGrus: A WiFi-based gesture recognition system using software-defined radio," *IEEE Access*, vol. 7, pp. 131102–131113, 2019.

[12] L. Sun, S. Sen, D. Koutsonikolas, and K.-H. Kim, "WiDraw: Enabling hands-free drawing in the air on commodity WiFi devices," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 77–89.

[13] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of WiFi signal based human activity recognition," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 65–76.

[14] W. Jiang, C. Miao, F. Ma, S. Yao, Y. Wang, Y. Yuan, H. Xue, C. Song, X. Ma, D. Koutsonikolas, W. Xu, and L. Su, "Towards environment independent device free human activity recognition," in *Proc. 24th Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2018, pp. 289–304.

[15] P. Hillyard, A. Luong, A. S. Abrar, N. Patwari, K. Sundar, R. Farney, J. Burch, C. Porucznik, and S. H. Pollard, "Experience: Cross-technology radio respiratory monitoring performance study," in *Proc. 24th Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2018, pp. 487–496.

[16] D. Zhang, J. Wang, J. Jang, J. Zhang, and S. Kumar, "On the feasibility of Wi-Fi based material sensing," in *Proc. 25th Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2019, pp. 41:1–41:16.

[17] J. Ding and R. Chandra, "Towards low cost soil sensing using Wi-Fi," in *Proc. 25th Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2019, pp. 39:1–39:16.

[18] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Keystroke recognition using WiFi signals," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 90–102.

[19] M. Zhao, F. Adib, and D. Katabi, "Emotion recognition using wireless signals," *Commun. ACM*, vol. 61, no. 9, pp. 91–100, Aug. 2018.

[20] K. Ohara, T. Maekawa, and Y. Matsushita, "Detecting state changes of indoor everyday objects using Wi-Fi channel state information," in *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, Sep. 2017, pp. 88:1–88:28.

[21] M. Kotaru and S. Katti, "Position tracking for virtual reality using commodity WiFi," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 68–79.

[22] J. Xiong, K. Sundaresan, and K. Jamieson, "ToneTrack: Leveraging frequency-agile radios for time-based indoor wireless localization," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 537–549.

[23] L. Shangguan, Z. Yang, X. A. Liu, Z. Zhou, and Y. Liu, "Relative localization of RFID tags using spatial-temporal phase profiling," in *Proc. 12th USENIX Symp. Netw. Syst. Design Implement.*, pp. 251–263, May 2015.

[24] Y. Ma, X. Hui, and E. C. Kan, "3D real-time indoor localization via broadband nonlinear backscatter in passive devices with centimeter precision," in *Proc. 22nd Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2016, pp. 216–229.

[25] M. Khaledi, M. Khaledi, S. Sarkar, S. Kasera, N. Patwari, K. Derr, and S. Ramirez, "Simultaneous power-based localization of transmitters for crowdsourced spectrum monitoring," in *Proc. 23rd Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2017, pp. 235–247.

[26] Y. Ma, N. Selby, and F. Adib, "Minding the billions: Ultra-wideband localization for deployed RFID tags," in *Proc. 23rd Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2017, pp. 248–260.

[27] D. Chen, K. G. Shin, Y. Jiang, and K.-H. Kim, "Locating and tracking BLE beacons with smartphones," in *Proc. 13th Int. Conf. Emerg. Netw. Exp. Technol.*, Nov. 2017, pp. 263–275.

[28] E. Soltanaghaei, A. Kalyanaraman, and K. Whitehouse, "Multipath triangulation: Decimeter-level WiFi localization and orientation with a single unaided receiver," in *Proc. 16th Annu. Int. Conf. Mobile Syst., Appl., Services*, Jun. 2018, pp. 376–388.

[29] T. Koike-Akino, P. Wang, M. Pajovic, H. Sun, and P. V. Orlik, "Fingerprinting-based indoor localization with commercial MMWave WiFi: A deep learning approach," *IEEE Access*, vol. 8, pp. 84879–84892, 2020.

[30] F. Adib, Z. Kabelac, and D. Katabi, "Multi-person localization via RF body reflections," in *Proc. 12th USENIX Symp. Netw. Syst. Design Implement.*, May 2015, pp. 279–292.

[31] K. Ohara, T. Maekawa, Y. Kishino, Y. Shirai, and F. Naya, "Transferring positioning model for device-free passive indoor localization," in *Proc. ACM Int. Joint Conf. Pervas. Ubiquitous Comput. (UbiComp)*, 2015, pp. 885–896.

[32] Y. Xie, J. Xiong, M. Li, and K. Jamieson, "MD-track: Leveraging multidimensionality for passive indoor Wi-Fi tracking," in *Proc. 25th Annu. Int. Conf. Mobile Comput. Netw.*, Aug. 2019, pp. 8:1–8:16.

[33] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proc. 19th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2013, pp. 27–38.

[34] H. Abdelnasser, K. Harras, and M. Youssef, "A ubiquitous WiFi-based fine-grained gesture recognition system," *IEEE Trans. Mobile Comput.*, vol. 18, no. 11, pp. 2474–2487, Nov. 2019.

[35] K. Joshi, D. Bharadia, M. Kotaru, and S. Katti, "WiDeo: Fine-grained device-free motion tracing using RF backscatter," in *Proc. 12th USENIX Conf. Netw. Syst. Design Implement.*, May 2015, pp. 189–204.

[36] M. Raja, Z. Vali, S. Palipana, D. G. Michelson, and S. Sigg, "3D head motion detection using millimeter-wave Doppler radar," *IEEE Access*, vol. 8, pp. 32321–32331, 2020.

[37] X. Zhuge and A. G. Yarovoy, "A sparse aperture MIMO-SAR-Based UWB imaging system for concealed weapon detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 1, pp. 509–518, Jan. 2011.

[38] F. Adib, C.-Y. Hsu, H. Mao, D. Katabi, and F. Durand, "Capturing the human figure through a wall," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–13, Nov. 2015.

[39] N. M. Iya, A. H. Muqaibel, and U. M. Johar, "Ultra wideband wall compensation for through-wall tracking and imaging," in *Proc. IEEE Int. Conf. Ultra-Wideband (ICUWB)*, Sep. 2011, pp. 81–85.

[40] N. Matsumoto, J. Kawasaki, M. Suzuki, S. Saruwatari, and T. Watanabe, "Crowdedness estimation using RSSI on already-deployed wireless sensor networks," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2019, pp. 1–7.

[41] F. Adib, H. Mao, Z. Kabelac, D. Katabi, and R. C. Miller, "Smart homes that monitor breathing and heart rate," in *Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst.*, Apr. 2015, pp. 837–846.

[42] B. Xie, J. Xiong, X. Chen, E. Chai, L. Li, Z. Tang, and D. Fang, "Tagtag: Material sensing with commodity RFID," in *Proc. 17th Conf. Embedded Netw. Sensor Syst.*, Nov. 2019, pp. 338–350.

[43] D. Vasisht, G. Zhang, O. Abari, H.-M. Lu, J. Flanz, and D. Katabi, "In-body backscatter communication and localization," in *Proc. Conf. ACM Special Interest Group Data Commun.*, Aug. 2018, pp. 132–146.

[44] C. Gao, Y. Li, and X. Zhang, "LiveTag: Sensing human-object interaction through passive chipless WiFi tags," in *Proc. 15th USENIX Symp. Netw. Syst. Design Implement.*, Apr. 2018, pp. 533–546.

[45] T. J. Pierson, T. Peters, R. Peterson, and D. Kotz, "Proximity detection with single-antenna IoT devices," in *Proc. 24th Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2018, pp. 21:1–21:15.

[46] T. Wei and X. Zhang, "Gyro in the air: Tracking 3D orientation of batteryless Internet-of-Things," in *Proc. 22nd Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2016, pp. 55–68.

[47] F. Adib and D. Katabi, "See through walls with Wi-Fi!" in *Proc. Conf. ACM Special Interest Group Data Commun.*, Aug. 2013, pp. 75–86.

[48] L. Yang, Q. Lin, X. Li, T. Liu, and Y. Liu, "See through walls with COTS RFID systems!" in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 487–499.

[49] F. Adib, Z. Kabelac, D. Katabi, and C. Robert Miller, "3D tracking via body radio reflections," in *Proc. 11th USENIX Conf. Netw. Syst. Design Implement.*, Apr. 2014, pp. 317–329.

[50] J. Wang, D. Vasisht, and D. Katabi, "RF-IDraw: Virtual touch screen in the air using RF signals," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 4, pp. 235–246, Oct. 2014.

[51] Z. Luo, Q. Zhang, Y. Ma, M. Singh, and F. Adib, "3D backscatter localization for fine-grained robotics," in *Proc. 16th USENIX Symp. Netw. Syst. Design Implement.*, Feb. 2019, pp. 765–781.

[52] N. Xiao, P. Yang, X. Li, Y. Zhang, Y. Yan, and H. Zhou, "MilliBack: Real-time plug-n-play millimeter level tracking using wireless backscattering," in *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, Sep. 2019, pp. 112:1–112:23.

[53] A. Mackey and P. Spachos, "LoRa-based localization system for emergency services in GPS-less environments," in *Proc. IEEE INFOCOM-IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 939–944.

[54] R. Ayyalasomayajula, D. Vasisht, and D. Bharadia, "BLoc: CSI-based accurate localization for BLE tags," in *Proc. 14th Int. Conf. Emerg. Netw. Exp. Technol.*, Dec. 2018, pp. 126–139.

[55] T. Murakami, M. Miyazaki, S. Ishida, and A. Fukuda, "Wireless LAN-based CSI monitoring system for object detection," *Electronics*, vol. 7, no. 11, pp. 290:1–290:11, Nov. 2018.

[56] USRP, Ettus Res., North Mopac Expressway, Austin, TX, USA.

[57] A. Khattab, J. Camp, C. Hunter, P. Murphy, A. Sabharwal, and E. W. Knightly, "WARP: A flexible platform for clean-slate wireless medium access protocol design," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 12, no. 1, pp. 56–58, Jan. 2008.

[58] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 1, p. 53, Jan. 2011.

[59] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity WiFi," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 53–64.

[60] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM CCR*, vol. 41, no. 1, p. 53, Jan. 2011.

[61] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity WiFi," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*. New York, NY, USA: ACM, 2015, pp. 53–64.

[62] E. Perahia and R. Stacey, *Next Generation Wireless LANs: 802.11n and 802.11ac*. Cambridge, U.K.: Cambridge Univ. Press, 2013.

[63] *IEEE Standard for Information Technology–Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks–Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012), Dec. 2016, pp. 1–3534.

[64] I. Goodfellow, J. P. Abadie, M. Mirza, B. Xu, D. W. Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, Dec. 2014, pp. 2672–2680.

[65] Z. Wang, Q. She, and T. E. Ward, "Generative adversarial networks in computer vision: A survey and taxonomy," *ACM Comput. Surv.*, vol. 54, no. 2, Feb. 2021, doi: 10.1145/3439723.

[66] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng, "Recent progress on generative adversarial networks (GANs): A survey," *IEEE Access*, vol. 7, pp. 36322–36333, Mar. 2019.

[67] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. 4th Int. Conf. Learn. Represent. (ICLR)*, 2016, pp. 1–16. [Online]. Available: http://arxiv.org/abs/1511.06434

[68] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 105–114.

[69] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, Dec. 2012, pp. 1097–1105.

[70] Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? A new look at signal fidelity measures," *IEEE Signal Process. Mag.*, vol. 26, no. 1, pp. 98–117, Jan. 2009.

[71] 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. (CVPR)*, Jun. 2016, pp. 779–788.

[72] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *CoRR*, vol. abs/1804.02767, pp. 1–6, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[73] *Yolov3,Weight*. [Online]. Available: https://pjreddie.com/media/files/yolov3.weights

[74] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," *CoRR*, vol. abs/1405.0312, pp. 1–15, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[75] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

SORACHI KATO (Student Member, IEEE) is currently pursuing the bachelor's degree with the School of Engineering, Osaka University, Japan. His research interest includes wireless sensing. He is a Student Member of IPSJ.

TAKERU FUKUSHIMA (Member, IEEE) received the B.E. and M.E. degrees from Osaka University, Japan, in 2018 and 2020, respectively. He is currently working with NTT Access Network Service Systems Laboratories. His current research interests include 5G new radio and other unlicensed wireless access networks. He is a member of IPSJ and IEICE.

TOMOKI MURAKAMI received the B.E., M.E., and Dr.Eng. degrees from Waseda University, Japan, in 2006, 2008, and 2015, respectively. In 2008, he joined NTT Network Innovation Laboratories, Nippon Telegraph and Telephone Corporation (NTT), Yokosuka, Japan. He is currently a Research Engineer with the Wireless Access Systems Project. His current research interest includes high-efficiency technologies for future wireless systems. He is a member of IEICE. He received the Young Engineer Award from the IEICE in 2010, the Active Research Award from IEICE Antenna and Propagation (AP) in 2010, the Best Tutorial Paper Award from IEICE Communications society in 2014, the Best Paper Award and the KIYASU-Zen'iti Award from the IEICE in 2015, and the Best Paper Award from IEICE on AP in 2016.

HIRANTHA ABEYSEKERA (Member, IEEE) received the B.Eng., M.Eng., and Ph.D. degrees in communications engineering from Osaka University, Japan, in 2005, 2007, and 2010, respectively. He joined NTT Network Innovation Laboratories, Yokosuka, Japan, in 2010, where he was involved in the research and development of next-generation wireless LAN systems. He is currently working as a Senior Research Engineer with NTT Access Service Systems Laboratories, NTT Corporation. His research interest includes resource allocation in wireless LANs. He received the IEEE VTS Japan Student Paper Award in 2009.
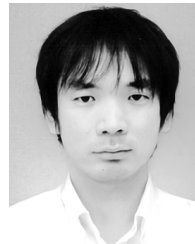
YUSUKE IWASAKI received the B.E. and M.E. degrees from Osaka University, Japan, in 2017 and 2020, respectively. He is a member of IPSJ and IEICE.

TAKASHI WATANABE (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Osaka University, Japan, in 1982, 1984, and 1987, respectively. He joined the Faculty of Engineering, Tokushima University, as an Assistant Professor, in 1987, and moved to the Faculty of Engineering, Shizuoka University, in 1990. He was a visiting Researcher with the University of California, Irvine, CA, USA, from 1995 to 1996. He has served on many program committees for networking conferences, IEEE, ACM, IPSJ, and The Institute of Electronics, Information and Communication Engineers, Japan (IEICE). He is currently a Professor with the Graduate School of Information Science and Technology, Osaka University. His research interests include mobile networking, ad hoc networks, sensornetworks, ubiquitous networks, intelligent transport systems, specially MAC, and routing. He is a member of IEEE Communications Society and IEEE Computer Society, as well as IPSJ and IEICE.

TAKUYA FUJIHASHI (Member, IEEE) received the B.E. and M.S. degrees from Shizuoka University, Japan, in 2012 and 2013, respectively, and the Ph.D. degree from the Graduate School of Information Science and Technology, Osaka University, Japan, in 2016. He has been an Assistant Professor with the Graduate School of Information Science and Technology, Osaka University, since April, 2019. His research interests include video compression and communications, with a focus on immersive video coding and streaming. He was a Research Fellow (PD) of the Japan Society for the Promotion of Science, in 2016. From 2014 to 2016, he was a Research Fellow (DC1) of the Japan Society for the Promotion of Science. From 2014 to 2015, he was an Intern with Mitsubishi Electric Research Laboratories (MERL) working with the Electronics and Communications Group.

SHUNSUKE SARUWATARI (Member, IEEE) received the Dr.Sci. degree from the University of Tokyo, in 2007. From 2007 to 2008, he was a visiting Researcher with the Illinois Genetic Algorithm Laboratory, University of Illinois at Urbana–Champaign. From 2008 to 2012, he was a Research Associate with the RCAST, University of Tokyo. From 2012 to 2016, he was an Assistant Professor with Shizuoka University. He has been an Associate Professor with Osaka University, since 2016. His research interests include wireless networks, sensor networks, and system software.

● ● ●