Signal Demodulation With Machine Learning Methods for Physical Layer Visible Light Communications: Prototype Platform, Open Dataset, and Algorithms

SHUAI MA[1,2]**, JIAHUI DAI**[1]**, SONGTAO LU**[3]**, (Member, IEEE), HANG LI**[4]**, (Member, IEEE), HAN ZHANG**[5]**, CHUN DU**[1]**, AND SHIYIN LI**[1]

[1]School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China
[2]State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China
[3]Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455, USA
[4]Shenzhen Research Institute of Big Data, Shenzhen 518172, China
[5]Department of Electrical and Computer Engineering, University of California at Davis, Davis, CA 95616, USA

Corresponding author: Shiyin Li (lishiyin@cumt.edu.cn)

ABSTRACT In this paper, we investigate the design and implementation of machine learning (ML)-based demodulation methods in the physical layer of visible light communication (VLC) systems. We build a flexible hardware prototype of an end-to-end VLC system, from which the received signals are collected as the real data. The dataset is available online, which contains eight types of modulated signals. Then, we propose three ML demodulators based on convolutional neural network (CNN), the deep belief network (DBN), and adaptive boosting (AdaBoost), respectively. Specifically, the CNN-based demodulator converts the modulated signals to images and recognizes the signals by the image classification. The proposed DBN-based demodulator contains three restricted Boltzmann machines to extract the modulation features. The AdaBoost method includes a strong classifier that is constructed by the weak classifiers with the k-nearest neighbor algorithm. These three demodulators are trained and tested by our online open dataset. The experimental results show that the demodulation accuracy of the three data-driven demodulators drops as the transmission distance increases. A higher modulation order negatively influences the accuracy for a given transmission distance. Among the three ML methods, the AdaBoost modulator achieves the best performance.

INDEX TERMS Visible light communication, machine learning, demodulation, CNN, DBN, AdaBoost.

I. INTRODUCTION

With the rapidly increasing number of mobile digital devices and the soaring high volume of wireless data traffic, the high speed wireless transmission is also highly demanded. Traditional radio frequency (RF) systems are currently facing spectrum crisis, which is the bottleneck of enhancing the network capacity [1]. Visible light communication (VLC), with advantages like huge unregulated spectrum, high security and immunity to electromagnetic interference, has sparked significant research attention as a promising solution for

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

short range wireless communications [2]. Through massive deployment of light-emitting diodes (LEDs), VLC typically employs the intensity modulation and direct detection (IM/DD) technique for both the illumination and data transmissions [3]–[9], where the information is represented by the real and nonnegative light signals.

Demodulation of radio signals plays a fundamental role in VLC systems. In general, the traditional demodulators could be categorized into two classes: coherent and noncoherent demodulators. Moreover, the priori knowledge, such as channel state information (CSI) or channel noise, is usually required. Most of previous works [10]–[12] indeed assume that each receiver can accurately estimate the fading

coefficients. In slow-fading scenarios, such CSI might be obtained via estimation from training sequences. However, in fast-fading scenarios, CSI is usually hard to estimate since the fading coefficients vary quickly within the period of one transmission block. Besides, most of existing works assume that the VLC channel suffers from additive white Gaussian noise (AWGN), and thus the applied demodulators are optimal in terms of the AWGN channel. However, the practical VLC channels are not easy to model since there exist too many factors, including but not limited to: limited modulation bandwidth of LEDs, multi-path dispersion, impulse noise, spurious or continuous jamming, and low sensitivity of commercial photodetector (PD). Even though the channel can be approximated by a complex model, the non-casual knowledge of the channel model might not be available at the receiver, especially when the channel fading is non-stationary with unknown distributions.

Given the above issues, machine learning (ML) based model-free demodulators become more attractive, where the requirements for the priori knowledge can be widely relaxed or even removed [13]. Huang *et al.* [14], [15] and Gui *et al.* [16] used neural networks which are considered as black boxes to detect the channel condition but with high computational complexity. Since the information of the modulated signals is represented by the amplitude and phase, feature extraction is critically important to the signal demodulation. Note that in conventional RF systems, ML based demodulators have been investigated, such as a neural network demodulator [17] and a one-dimensional convolutional neural network (CNN) based demodulator [18] for binary phase shift keying (BPSK) signals. Also, a deep convolutional neural network (DCNN) demodulator was proposed in [19] to respectively demodulate symbol sequences from mixed signals. Fan and Wu [20] showed that the deep belief network (DBN) based demodulator is feasible for the AWGN channel with certain channel impulse response and the Rayleigh non-frequency-selective flat fading channel. In [21], a deep learning (DL) based detection method was proposed for signal demodulation in short range multipath channel without any channel equalization.

Different from RF communication systems, the transmitted signals of VLC should be real and non-negative due to the IM/DD mechanism. In the research of VLC systems, the ML based approaches have been investigated to some extent. In [22], a DL based autoencoder was designed for multi-dimensional color modulation in multicolored VLC systems, which can reduce the average symbol error probability. A soft binarization training strategy was proposed for autoencoder VLC systems in [23], which yielded an efficient on-off keying (OOK) transceiver over general optical channels. However, the existing works [17]–[23] are based on synthetic data rather than real datasets. To the best of our knowledge, the ML based demodulation schemes have not been well studied in VLC systems, and there is no open real measurement data yet.

In this paper, we present a unified data-driven framework of demodulation by ML approaches. To be specific, we propose three data-driven demodulation methods: CNN, DBN, adaptive boosting (AdaBoost) [24] based demodulators for end-to-end VLC systems. Also, the performance of the three data-driven demodulators are evaluated for the different modulation schemes via the real measured data. Our main contributions are as follows:

- We propose a flexible end-to-end VLC system hardware prototype to study data-driven demodulation approaches. By exploiting this prototype, we collect received signal data in real physical environments in eight modulation schemes, i.e., OOK, quadrature phase shift keying (QPSK), 4-pulse position modulation (PPM), 16-quadrature amplitude modulation (QAM), 32-QAM, 64-QAM, 128-QAM and 256-QAM. We establish an open online real modulated dataset available at https://pan.baidu.com/s/1rS143bEDaOTEi CneXE67dg, where the transmission distance of the eight modulated signals is measured from 0cm to 140cm. To the best of our knowledge, this is the first open real modulated signals dataset of VLC systems.
- Three ML-based demodulators are designed. We propose a CNN based demodulator with two convolutional layers and two pooling layers. It first converts the modulated signals to images and then identifies the signals by the image classification. Then we develop a DBN based demodulator with three restricted Boltzmann machines (RBMs) to extract the modulation features. Finally, an AdaBoost based demodulator is presented, where a strong classifier is constructed by several weak classifiers with the k-nearest neighbor (KNN) algorithm.
- Based on the established real dataset, we investigate the demodulation performance of the proposed three data-driven demodulators. Specifically, the demodulation accuracy of the three ML based demodulators is decreasing over the transmission and the modulation order for a fixed transmission distance. Experimental results also show that the demodulation accuracy of the AdaBoost based demodulators is higher than other demodulators. Moreover, for the short distance or high SNR scenario, a high-order modulation is preferred.

Notation: The following notations are used throughout the paper. Bold upper case letters represent matrices, e,g, $\mathbf{A}$. Bold lower case letters represent vectors, e,g, $\mathbf{a}$. $[\cdot]^T$ means transpose, and Re $[\cdot]$ is used to obtain the real part. $[\cdot]_{p,q}$ indicates the element at the pth row and the qth column. Moreover, $[\cdot]_p$ indicates the pth element. $\|\cdot\|_2$ is the L_2 norm operator, and $\mathbb{R}$ is the real number sets. The natural logarithm $\ln(\cdot)$ is used. $\approx$ means approximate equals, and $\sim$ means subjecting to certain distribution. ∂ denotes the partial derivation and $*$ is the convolution operator. $\leftarrow$ means that the values on the left is updated by the values on the right.
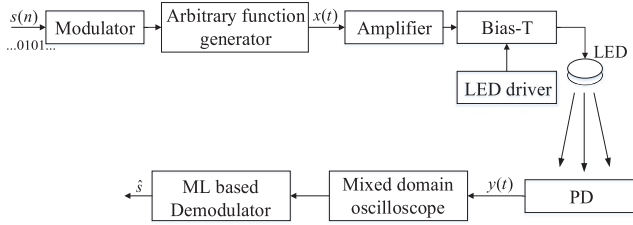
FIGURE 1. The VLC system model with ML based demodulator.

II. SYSTEM MODEL

As illustrated in Fig. 1, we propose a flexible end-to-end VLC prototype, which consists a modulation block, an arbitrary function generator, an amplifier, a bias-T, a LED driver, a single LED, a single PD, a mixed domain oscilloscope, and a ML based demodulation block. According to Fig. 1, the digital signal $s(n)$ is modulated by the M-QAM scheme, converted to the analog signal by the arbitrary function generator, and further amplified by the amplifier. After amplification, the signal adds the direct current (DC) at the Bias-T. Finally, the signal is transformed to the visible light by LED, and sent out to the wireless channels. At the receiver, the optical signal from LED is converted to the analog signal through PD, and then the analog signal is converted to a digital signal at the mixed domain oscilloscope. Afterwards, the digital signal is demodulated by the ML based demodulator.

By exploiting digital modulation schemes, such as M-QAM and M-PPM, the transmitted signal $x(t)$ is given as

$$x(t) = \text{Re}\left[s(t) p(t) e^{j2\pi f_c t} \right], \quad 0 < t \le T, \quad (1)$$

where $s(t)$ denotes the baseband signal, f_c denotes the carrier frequency, $p(t)$ stands for the signal pulse, and T represents the period of signal.

Let g denote the channel gain between LED and PD, which includes both line-of-sight (LOS) path and multi-reflection paths. At the PD side, the received signal $y(t)$ is given as:

$$y(t) = gx(t) + n(t), \quad (2)$$

where $n(t)$ is the received noise. Via the mixed domain oscilloscope, the received analog signal $y(t)$ is sampled to digital signals.

Let $\hat{\mathbf{y}}_i = [\hat{y}_{(i-1)N+1}, \hat{y}_{(i-1)N+2}, \dots, \hat{y}_{iN}]^T$ denote the received signal vector during the ith period, where $\hat{y}_{(i-1)N+n}$ is the nth sampled point $\hat{y}_{(i-1)N+n} = y\left(\frac{n-1}{N}T + (i-1)T\right)$, and N is the number of samples during one period. Assume that training data set contains K periods of sampled vectors and $1 \le i \le K$. Before the demodulator processing, the received data $\{\hat{\mathbf{y}}_i\}_{i=1}^K \triangleq \{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_{NK}\}$ is normalized to [0, 1], which can significantly reduce the calculation time of ML [25]. The normalized sample $\bar{y}_i$ is given by

$$\bar{y}_i = \frac{\hat{y}_i - \hat{y}_{\min}}{\hat{y}_{\max} - \hat{y}_{\min}}, \quad 1 \le i \le NK, \quad (3)$$

where $\hat{y}_{\min} = \min\limits_{1 \le i \le NK} \hat{y}_i$, $\hat{y}_{\max} = \max\limits_{1 \le i \le NK} \hat{y}_i$.
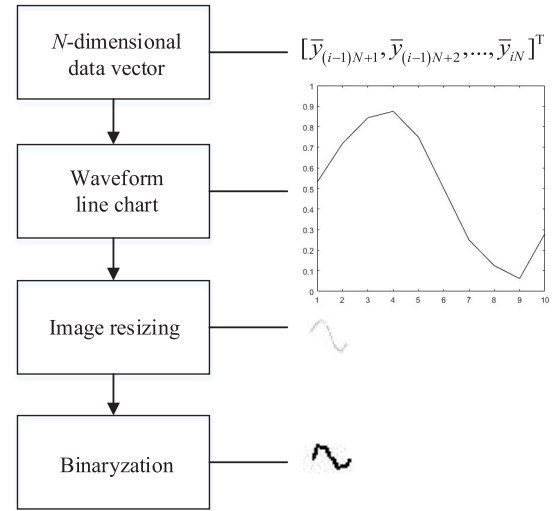

FIGURE 2. The visualization block.

After normalization, we use

$$\bar{\mathbf{y}}_i = [\bar{y}_{(i-1)N+1}, \bar{y}_{(i-1)N+2}, \dots, \bar{y}_{iN}]^T \quad (4)$$

to denote the ith normalized signal vector. Moreover, let z_i denote the label for the normalized vector $\bar{\mathbf{y}}_i$ and $\mathcal{C}$ the label set, i.e., $z_i \in \mathcal{C}$ for $i = 1, 2, \dots, K$. The label set $\mathcal{C}$ is determined by the modulation scheme. For example, $\mathcal{C} = \{1, 2, 3, 4\}$ is used for quadrature phase shift keying QPSK signal. Let $\mathcal{T} = \{(\bar{\mathbf{y}}_1, z_1), (\bar{\mathbf{y}}_2, z_2), \dots, (\bar{\mathbf{y}}_K, z_K)\}$ denote the labeled dataset.

In the following sections, we propose three ML based demodulators and present their structure in details.

III. CNN BASED DEMODULATOR

Due to the sparse connectivity and parameter sharing characteristics, CNN has a simple structure and strong adaptability and is applied in various domains [26], [27]. For single carrier modulation, the amplitude and phase information of signal can be extracted for classification. Therefore, we investigate the CNN based demodulator, which includes a visualization block and a CNN network. We first convert the data vector $\bar{\mathbf{y}}_i$ into a two-dimensional image format so that the CNN based demodulator can interpret the data as images. Specifically, as shown in Fig. 2, the elements of $\bar{\mathbf{y}}_i$ are first transformed to a point on the two-dimensional plane. First, we consider n as the coordinate of horizontal axis and the value of $\bar{y}_{(i-1)N+n}$ as the coordinate of vertical axis, and transform the vector to N points. Then, we connect these points by polylines, so that we can obtain the waveform with horizontal axis range of $[1, N]$ and vertical axis range of [0, 1]. In waveform images, both amplitude and phase information of the modulated signals are represented by waveforms with high pixel density. To reduce the computational load of computer and preserve the useful information, we resize the grey image with less pixels by applying the bicubic interpolation algorithm. Also, the resized grey image
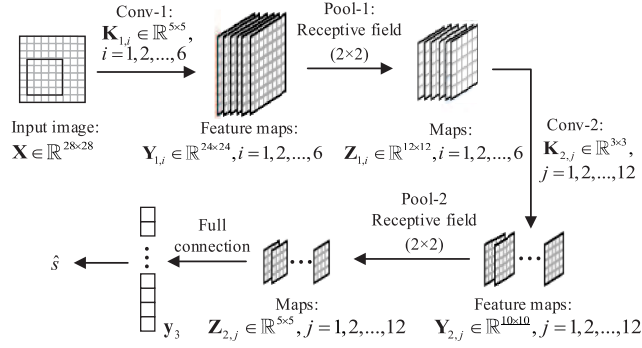
FIGURE 3. The structure of CNN.

TABLE 1. Parameters setting of CNN.

Layer	Kernel size	Stride	Output size
Input			28×28
Conv-1	5×5	1	24×24
Pool-1	2×2	2	12×12
Conv-2	3×3	1	10×10
Pool-2	2×2	2	5×5

FIGURE 4. The max-pooling operation with 2 × 2 filter and stride 2.

is changed into a binary image by the global thresholding algorithm [28], which can further distinguish the waveform from the background. Finally, we obtain the output image matrix $\mathbf{X}$ with a size 28×28, i.e, $\mathbf{X} \in \mathbb{R}^{28 \times 28}$.

Then, the output $\mathbf{X}$ of the visualization block is processed by the presented CNN network, which includes two convolutional layers, two pooling layers, and one full-connected layer as shown in Fig. 3. Let Conv-1 and Conv-2 stand for the first and second convolutional layer, respectively. Moreover, the Pool-1 and Pool-2 denote the first and second pooling layer, respectively.

As shown in Fig. 3, the input image first convolutes with six kernels in Conv-1, respectively. Then, the Conv-1 outputs six feature maps. In Pool-1, the feature maps are compressed to maps by the (2×2) receptive field [26]. Then, the maps are processed via kernels, and further compressed to maps by the (2×2) receptive field in Pool-2. Finally, the output maps of Pool-2 are connected via full connection to the output layer, whose dimension is determined by the modulation scheme.

The parameters of the CNN are shown in Table 1. Let $\mathbf{K}_{1,i}$ represent the ith kernel of Conv-1, $\mathbf{K}_{1,i} \in \mathbb{R}^{5 \times 5}$, $i = 1, 2, \ldots, 6$. Moreover, let $\mathbf{Y}_{1,i}$ denote the output feature map obtained by $\mathbf{K}_{1,i}$, which can be expressed by [29]

$$\mathbf{Y}_{1,i,p,q} = \text{sigmoid}\left(b_i + \left[\mathbf{X} * \mathbf{K}_{1,i}\right]_{p,q}\right), \quad (5)$$

where b_i stands for the bias of $\mathbf{K}_{1,i}$, $p = 1, 2, \ldots, 24$, $q = 1, 2, \ldots, 24$. Here, we choose sigmoid $(x) \triangleq \frac{1}{1+e^{-x}}$ as the activation function.

The convolutional layer is followed by Pool-1, which is used for down sampling of the output feature maps and increasing the robustness of the model. The pooling method used in this paper is max-pooling, as shown in Fig. 4. The maximum value in a submatrix of size 2×2 is treated as the local output. Let $\mathbf{Z}_{1,i}$ stand for the pooling result of $\mathbf{Y}_{1,i}$, and it can be expressed by

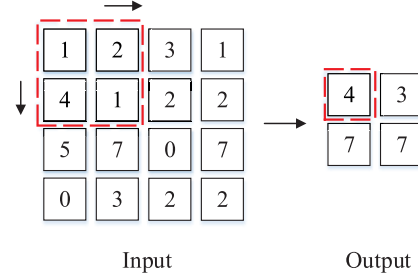$$\mathbf{Z}_{1,i} = \text{pooling}\left(\mathbf{Y}_{1,i}\right), \quad (6)$$

where pooling $(\cdot)$ stands for the component-wise max-pooling function.

Let $\mathbf{K}_{2,j}$ stand for the kernel adopted in Conv-2, $\mathbf{K}_{2,j} \in \mathbb{R}^{3 \times 3}$, $j = 1, 2, \ldots, 12$. Assume that $\mathbf{Y}_{2,j}$ is the output feature

map of $\mathbf{K}_{2,j}$, $\mathbf{Y}_{2,j} \in \mathbb{R}^{10 \times 10}$, $j = 1, 2, \ldots, 12$. It can be obtained by

$$\mathbf{Y}_{2,j,p,q} = \text{sigmoid}\left(b_j + \sum_i \left[\mathbf{Y}_{1,i} * \mathbf{K}_{2,j}\right]_{p,q}\right), \quad (7)$$

where $p = 1, 2, \ldots, 10$, $q = 1, 2, \ldots, 10$.

After Pool-2 with receptive field of 2×2, the output maps $\mathbf{Z}_{2,j}$ are transformed into a one-dimensional label space by the full-connected layer. Let $\mathbf{y}_3$ stand for the one-dimensional vector, the output label $\hat{z}$ can be expressed by

$$\hat{z} = \arg \max_i \left[\mathbf{y}_3\right]_i. \quad (8)$$

The dimension of $\mathbf{y}_3$ is determined by the modulation scheme employed. Then, the label $\hat{z}$ corresponds to the demodulation result $\hat{s}$.

IV. DBN BASED DEMODULATOR

DBN has been widely applied to address many practical problems such as handwritten recognition, speech recognition, and image classification, since it can efficiently extracts high-level and hierarchical features from the measured signal data by a multiple nonlinear transformation. RBM is the fundamental block of DBN, which is a realization of undirected graphical model and contains a layer of visible neurons and a layer of hidden neurons [30]. It is noted that there are only connections between the visible layer and the hidden layer.

Consider a DBN with three RBMs, as shown in Fig. 5. The first RBM is consisted of a visible layer $\mathbf{v} = [v_1, v_2, \ldots, v_m]^T$ and a hidden layer $\mathbf{h} = [h_1, h_2, \ldots, h_n]^T$, which contains m neurons and n neurons in the visible layer and hidden layer, respectively. Let $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n]^T$ denote the connection weight matrix between $\mathbf{v}$ and $\mathbf{h}$, where $\mathbf{w}_j = [w_{j1}, w_{j2}, \ldots, w_{jm}]^T$, $j = 1, 2, \ldots, n$. Moreover, $\mathbf{a} = [a_1, a_2, \ldots, a_m]^T$ and $\mathbf{b} = [b_1, b_2, \ldots, b_n]^T$ denote the bias of $\mathbf{v}$ and $\mathbf{h}$, respectively.
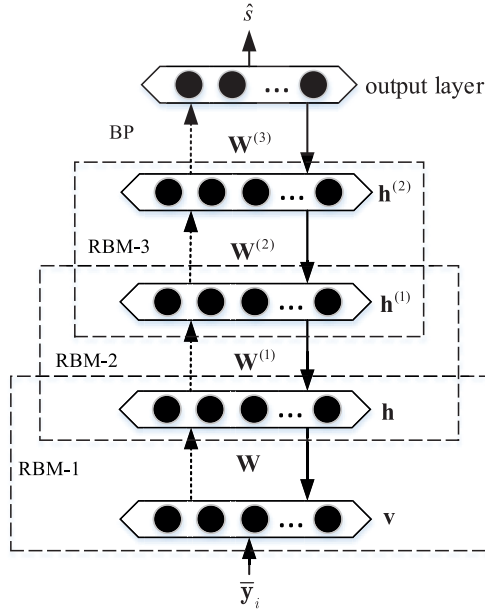
FIGURE 5. The structure of DBN.

RBM is an energy based model, which defines the probability distribution of variables by the energy function. With the normalized signal $\bar{\mathbf{y}}_i$ in $\mathcal{T}$, the energy of the first RBM is given by

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{a}^T \mathbf{v} - \mathbf{b}^T \mathbf{h} - \mathbf{h}^T \mathbf{W} \mathbf{v}, \qquad (9)$$

where $\mathbf{v} = \bar{\mathbf{y}}_i$. The probability distribution of the visible layer $\mathbf{v}$ is given by

$$p(\mathbf{v}) = \frac{1}{Z} \sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}, \qquad (10)$$

where $Z = \sum_{\mathbf{v}, \mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}$ is a normalization constant.

Then, the optimal parameters $\mathbf{W}, \mathbf{a}, \mathbf{b}$ can be obtained by maximizing the log-likelihood function as follows

$$\max_{\mathbf{W}, \mathbf{a}, \mathbf{b}} \sum_{\{\mathbf{v}\}} \ln p(\mathbf{v}). \qquad (11)$$

To solve the unconstrained optimization problem (11), we simply adopt the gradient descent method. The partial derivations with respect to variables $\mathbf{W}$, $\mathbf{a}$, and $\mathbf{b}$ can be respectively approximated by

$$\frac{\partial \ln p(\mathbf{v})}{\partial w_{ji}} \approx p(h_j = 1 \,|\, \mathbf{v}) v_i - p(h_j = 1 \,|\, \hat{\mathbf{v}}) \hat{v}_i, \quad (12a)$$

$$\frac{\partial \ln p(\mathbf{v})}{\partial a_i} \approx v_i - \hat{v}_i, \qquad (12b)$$

$$\frac{\partial \ln p(\mathbf{v})}{\partial b_j} \approx p(h_j = 1 \,|\, \mathbf{v}) - p(h_j = 1 \,|\, \hat{\mathbf{v}}), \qquad (12c)$$

where $p(h_j = 1 \,|\, \mathbf{v})$ and $p(h_j = 1 \,|\, \hat{\mathbf{v}})$ denote the conditional probability distribution of hidden neurons $\mathbf{h}$ given $\mathbf{v}$ and $\hat{\mathbf{v}}$, respectively. $\hat{\mathbf{v}} = [\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_m]^T$ denotes the reconstruction of visible states, which can be obtained as follows [31].

Given the visible layer $\mathbf{v}$, $p(h_j = 1 \,|\, \mathbf{v})$ is given by

$$p(v_i = 1 \,\big|\, \hat{\mathbf{h}}) = \mathrm{sigmoid}\left(a_i + \sum_{j=1}^{m} \hat{h}_j w_{ji}\right). \qquad (13)$$

Then, we can generate $\hat{\mathbf{h}} = [\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_n]^T$ according to distribution (13) as the following:

$$\hat{\mathbf{h}} \sim p(\mathbf{h} \,|\, \mathbf{v}). \qquad (14)$$

Similarly, the distribution of the visible layer $\mathbf{v}$ is given by

$$p(v_i = 1 \,\big|\, \hat{\mathbf{h}}) = \mathrm{sigmoid}\left(a_i + \sum_{j=1}^{m} \hat{h}_j w_{ji}\right). \qquad (15)$$

Then, the reconstructed data $\hat{\mathbf{v}}$ is generated based on distribution (15) as the following:

$$\hat{\mathbf{v}} \sim p(\mathbf{v} \,\big|\, \hat{\mathbf{h}}). \qquad (16)$$

Furthermore, the variables $\mathbf{W}, \mathbf{a}, \mathbf{b}$ are respectively updated as the following rules:

$$\mathbf{W} \leftarrow \mathbf{W} + \varepsilon \Delta \mathbf{W}, \qquad (17a)$$

$$\mathbf{a} \leftarrow \mathbf{a} + \varepsilon \Delta \mathbf{a}, \qquad (17b)$$

$$\mathbf{b} \leftarrow \mathbf{b} + \varepsilon \Delta \mathbf{b}, \qquad (17c)$$

where ε denotes the learning rate, $\Delta \mathbf{W}$, $\Delta \mathbf{a}$ and $\Delta \mathbf{b}$ are the partial gradients of the objective function with respect to $\mathbf{W}$, $\mathbf{a}$ and $\mathbf{b}$, respectively, as calculated in (12). By exploiting the gradient descent method, we obtain the optimal parameters $\hat{\mathbf{W}}$, $\hat{\mathbf{a}}$ and $\hat{\mathbf{b}}$ for the first RBM.

Then, the hidden layer $\mathbf{h}$ of the first RBM can be viewed as the visible layer of the second RBM, whose hidden layer is denoted as $\mathbf{h}^{(1)}$. After training the weight matrix and bias of the second RBM, $\mathbf{h}^{(1)}$ and $\mathbf{h}^{(2)}$ are viewed as the visible layer and hidden layer of the third RBM, respectively. After the third RBM is trained, all the parameters (weights and biases) of the RBMs are fine-tuned by a supervised back-propagation (BP) algorithm [32]. After training, the parameters of the DBN model are updated to approach the optimal classifier. The DBN is applied to demodulate signals at the test phase, where the demodulation results $\hat{s}$ is corresponding to classification result $\hat{z}$.

V. ADABOOST BASED DEMODULATOR

AdaBoost algorithm is a powerful tool that can integrate multiple independent weakly classifiers into a high-performance stronger classifier. In this paper, we exploit the AdaBoost method to demodulate signals, where the generation process of strong classifier is shown in Fig. 6. Here, KNN is employed as the weak classifier.

Suppose that the strong classifier is composed of Q KNNs [33]. For the qth KNN, the weight of samples in $\mathcal{T}$ is represented by $\mathbf{d}_q = [d_{q,1}, d_{q,2}, \ldots, d_{q,K}]^T$, $q = 1, 2, \ldots, Q$, and $d_{q,i}$ stands for the weight of the ith sample in $\mathcal{T}$. When $q = 1$, $d_{q,i} = 1/K$, $i = 1, 2, \ldots, K$. The training set of the qth KNN is represented by $\mathcal{T}_q$, which is
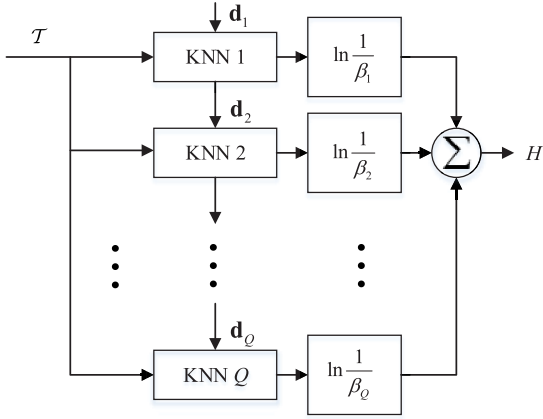
FIGURE 6. The generation process of the strong classifier.

FIGURE 7. The devices of the end-to-end VLC system prototype.

generated by re-sampling of $\mathcal{T}$ according to $\mathbf{d}_q$ [34]. Assume that $\mathcal{T}_q = \{(\mathbf{x}_{q,1}, z_{q,1}), (\mathbf{x}_{q,2}, z_{q,2}), \ldots, (\mathbf{x}_{q,K}, z_{q,K})\}$, and $(\mathbf{x}_{q,i}, z_{q,i}) \in \mathcal{T}$. The testing set is $\mathcal{T}$. $\tilde{\mathbf{y}}_i$ stands for the nearest sample of $\bar{\mathbf{y}}_i$ in training set $\mathcal{T}_q$, i.e.,

$$\tilde{\mathbf{y}}_i = \arg \min_{\{\mathbf{x}_{q,i}\}_{i=1}^{K}} \left\| \mathbf{x}_{q,i} - \bar{\mathbf{y}}_i \right\|_2, \tag{18}$$

where $\left\| \mathbf{x}_{q,i} - \bar{\mathbf{y}}_i \right\|_2$ is the Euclidean distance between $\mathbf{x}_{q,i}$ and $\bar{\mathbf{y}}_i$. Assume that the label of $\tilde{\mathbf{y}}_i$ is $\tilde{z}_i$, the KNN classifier categorizes $\bar{\mathbf{y}}_i$ to $\tilde{z}_i$. Hence, the classifier can be represented by $G_q(\bar{\mathbf{y}}_i) = \tilde{z}_i$, which means the classification result of the qth KNN for sample $\bar{\mathbf{y}}_i$ is $\tilde{z}_i$.

The error of G_q is defined as weighted sum of weights of the misclassified samples [35]:

$$e_q = \sum_{i=1}^{K} d_{q,i} \left(1 - I\left(G_q(\bar{\mathbf{y}}_i), z_i \right) \right), \tag{19}$$

where $I(a, b)$ is indication function:

$$I(a, b) = \begin{cases} 1, & \text{if } a = b, \\ 0, & \text{if } a \neq b. \end{cases}$$

Similarly, let $\mathbf{d}_{q+1} = \left[d_{q+1,1}, d_{q+1,2}, \ldots, d_{q+1,K} \right]^{\mathrm{T}}$ stand for the weight of samples for the $q + 1$th KNN, and it can be obtained by:

$$d_{q+1,i} = d_{q,i} e^{\ln \frac{1}{\beta_q} \left(1 - I(G_q(\bar{\mathbf{y}}_i), z_i) \right)}, \quad i = 1, 2, \ldots, K, \tag{20}$$

where β_q is computed as a function of e_q such that $\beta_q = \frac{e_q}{1 - e_q}$. Under the constraints of $e_q < 0.5$, $\beta_q < 1$. If $\bar{\mathbf{y}}_i$ is correctly classified, we have $I(G_q(\bar{\mathbf{y}}_i), z_i) = 1$, $d_{q+1,i} = d_{q,i}$. If $\bar{\mathbf{y}}_i$ is misclassified, $I(G_q(\bar{\mathbf{y}}_i), z_i) = 0$, and $d_{q+1,i}(i) = \frac{d_{q,i}}{\beta_q}$.

We redefine $d_{q+1,i}$ by the following normalization formula:

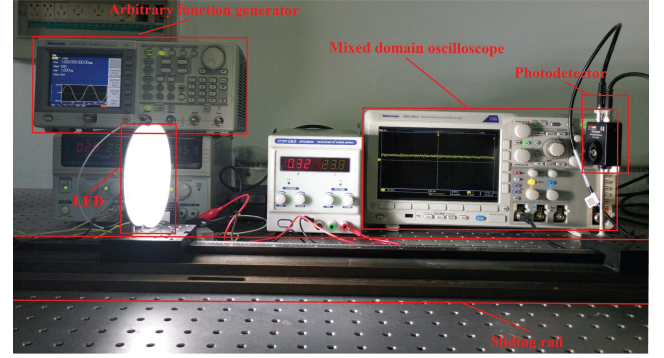$$d_{q+1,i} = \frac{d_{q+1,i}}{\sum\limits_{k=1}^{K} d_{q+1,k}}. \tag{21}$$

After generating Q KNNs, the strong classifier is determined by:

$$H(\mathbf{y}) = \hat{z} = \arg \max_{z \in C} \sum_{q=1}^{Q} \ln \frac{1}{\beta_q} I\left(G_q(\mathbf{y}), z \right), \tag{22}$$

where $\mathbf{y}$ denotes the test sample, $\ln \frac{1}{\beta_q}$ is the coefficient of G_q. $I(G_q(\mathbf{y}), z)$ can be treated as the voting value, i.e.: if $I(G_q(\mathbf{y}), z) = 1$, G_q classifies sample $\mathbf{y}$ into class z, otherwise $\mathbf{y}$ does not belong to class z. The class with the maximum sum of weighted voting value $\ln \frac{1}{\beta_q} I(G_q(\mathbf{y}), z)$ for all classifiers is identified as the classification result $\hat{z}$ of the AdaBoost classifier, and then $\hat{z}$ is mapped to demodulation result $\hat{s}$.

VI. EXPERIMENT RESULTS AND DISCUSSIONS

A. THE END-TO-END VLC SYSTEM PROTOTYPE

As shown in Fig. 7, the proposed end-to-end VLC system prototype includes a source computer, an arbitrary function generator, an amplifier, a bias-T, a LED, a sliding rail, a PD, and a mixed domain oscilloscope. We use this prototype to generate the real VLC modulation dataset and verify the proposed data-driven demodulation methods. The parameters of the devices used in the end-to-end VLC system prototype are listed in Table 2.

In the experiments, a serial binary bit stream is randomly generated and modulated in 8 different types of signals on computer with MATLAB. We sample N points in one period to generate modulated digital signals for each scheme, which is transferred to analog waveforms by the arbitrary function generator. The modulated current after amplification is superimposed on LED. At the receiver, the sampled digital signals are monitored and shown by the mixed domain oscilloscope. After normalization, we treat the signal in one period as input of DBN for training and testing. In CNN, we transfer the vector into image as demonstrated in Section III. The vector is considered as the feature of transmitted symbol and processed by AdaBoost so that it can be demodulated.

TABLE 2. Devices and parameters of the VLC system prototype.

Device/Parameter	value
Arbitrary function generator	Tektronix AFG3152C
Sampling rate	2500000 samples/second
Amplifier	Mini-Circuits ZHL-6A-S+
Gain of amplifier	25dB
Bias-T	SHWBT-006000-SFFF
PD	PDA10A-EC
Field of view (FOV) of PD	90°
Responsivity of PD	0.44A/W at 750nm
Mixed domain oscilloscope	Tektronix MDO3012
Power of LED	7.35W[1]
Half-intensity radiation angle	60°

TABLE 3. The structure and size of the dataset.

Modulation	N			
	10	20	40	80
OOK	72000	36000	18000	18000
QPSK	72000	36000	18000	18000
4-PPM	90000	45000	22500	11250
16-QAM	67500	33750	18000	18000
32-QAM	81000	36000	36000	36000
64-QAM	81000	72000	72000	72000
128-QAM	81000	72000	72000	72000
256-QAM	81000	72000	72000	72000

Our open dataset[2] contains eight modulation types, i.e., OOK, QPSK, 4-PPM, 16-QAM, 32-QAM, 64-QAM, 128-QAM and 256-QAM. For each type of modulation, there are four different numbers of sample points in each period, i.e., $N = 10, 20, 40, 80$. The number of periods in each case is listed in Table 3. Specially, $N = 8, 16, 32, 64$ for 4-PPM. Let d denote the distance between LED and PD. The data is collected for every 5cm from $d = 0$cm to $d = 140$cm and normalized. The illuminance of the ambient light is about 85 Lux. At the distance of $d = 100$cm, the illuminance of the LED is 492 Lux. Our database is available at https://pan.baidu.com/s/1rS143bEDaOTEiCneXE67dg. Eight modulation schemes are tested in experiments, where the numbers of signal periods for training and testing are listed in Table 4. For the DBN demodulator, we adopt the gradient descent method in pre-training stage. Then, the parameters are fine-tuned by the BP algorithm [32]. For the CNN demodulator, the BP algorithm is also used to train parameters.

B. EXPERIMENT RESULTS

The DBN used in the experiments consists of 10, 20, 40 and 80 visible units according to the dimension of the input data, and the size of output layer is determined by the demodulation scheme used. There are three hidden layers, and the size of each hidden layer and training parameters are listed in Table 5. For OOK signals, the three hidden layers have 10, 10, and 20 hidden units respectively. For 256-QAM signals,

[2]The dataset is collected in real physical environment, and the channel suffers from many factors such as limited LED bandwidth, multi-reflection, spurious or continuous jamming, etc.

TABLE 4. Training and testing dataset.

Modulation	Number of signal periods	
	Training	Testing
OOK	12000	6000
QPSK	12000	6000
4-PPM	7500	3750
16-QAM	12000	6000
32-QAM	24000	12000
64-QAM	48000	24000
128-QAM	48000	24000
256-QAM	48000	24000

TABLE 5. DBN structure and parameters.

Size of hidden layer-1	$10 \sim 500$
Size of hidden layer-2	$10 \sim 500$
Size of hidden layer-3	$20 \sim 2000$
Pre-training epoch	$50 \sim 1000$
BP epoch	$50 \sim 1000$
Batch size	100
Learning rate	0.1

there are 500, 500, and 2000 hidden units of the three layers. As for the CNN based demodulator, the batch size is 100 and the epoch number is $100 \sim 200$.

All the proposed methods are implemented with MATLAB R2016b and executed on a computer with an Intel Core i7-7700 CPU @ 3.60 GHz/32 GB RAM. The DeepLearnToolbox [36] is used to implement the CNN and DBN based classifiers. We first investigate the performance of the proposed CNN, DBN, and AdaBoost based demodulation methods versus distance d with $N = 40$. After training, the accuracies on test set is calculated. Moreover, both the support vector machine (SVM) based and the maximum likelihood (MLD) based demodulation methods are used for comparison. SVM is a supervised learning method which solve binary classification problems. In this paper, we combine SVMs to demodulate by one-to-one way. MLD classification is one of the supervised classification algorithms based on the Bayesian criterion, which assumes that the input feature vector follows N-dimensional normal distribution, and calculate the attribution probability of the input vector belonging to each category. The data vector is categorized to the class with the maximum attribution probability.

Fig. 8 (a), (b) and (c) show the demodulation accuracies of symbols of OOK, 32-QAM and 256-QAM modulated signals versus distance d, respectively. We can see that the demodulation accuracy of all methods decreases as the distance d increases. Specifically, Fig. 8 (a) shows that the demodulation accuracy of all methods of OOK modulated signals are close to 100% for $d \leq 70$cm; and for 70cm $< d \leq 140$cm, the proposed AdaBoost based demodulation method significantly outperforms other demodulation methods. Fig. 8 (b) shows that the demodulation accuracies of the 32-QAM modulated signals by all methods are close to 100% for $d \leq 40$cm. For 40cm $< d \leq 140$cm, the demodulation accuracy of the AdaBoost based demodulation method is the
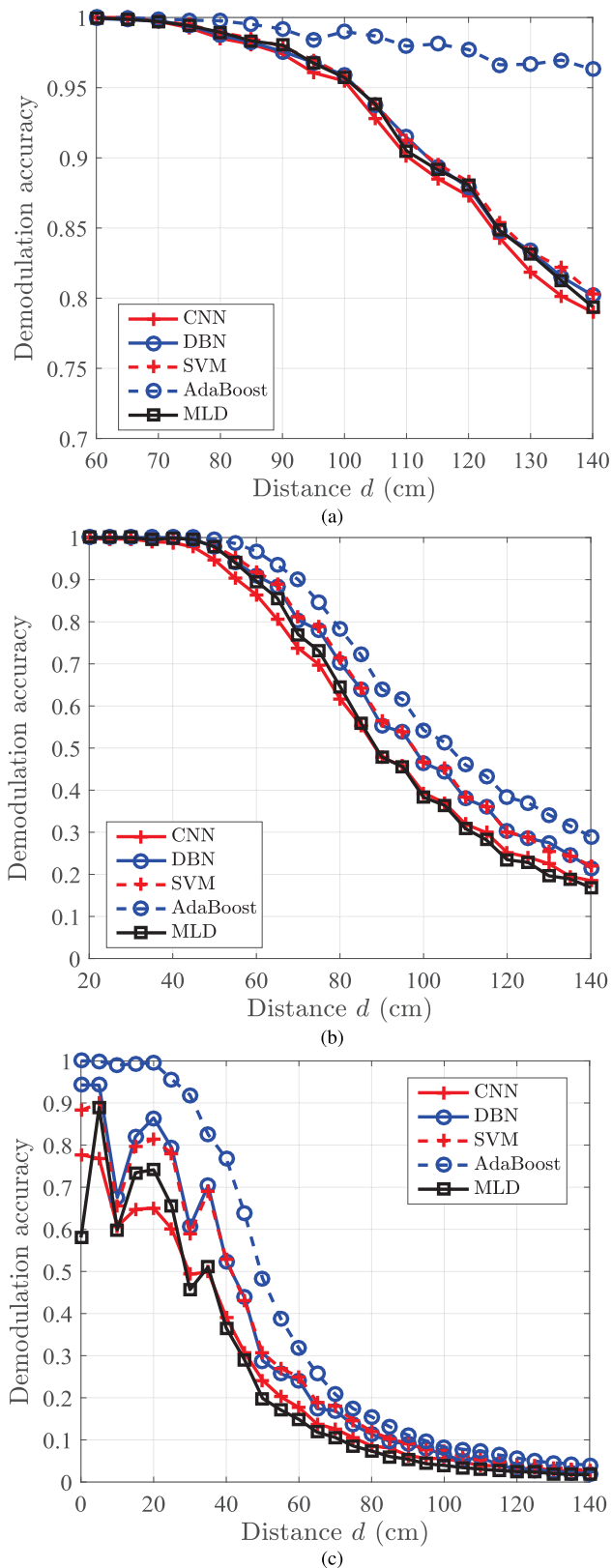
FIGURE 8. (a) The demodulation accuracy of OOK modulated signals versus distance *d* when *N* = 40. (b) The demodulation accuracy of 32-QAM modulated signals versus distance *d* when *N* = 40. (c) The demodulation accuracy of 256-QAM modulated signals versus distance *d* when *N* = 40.
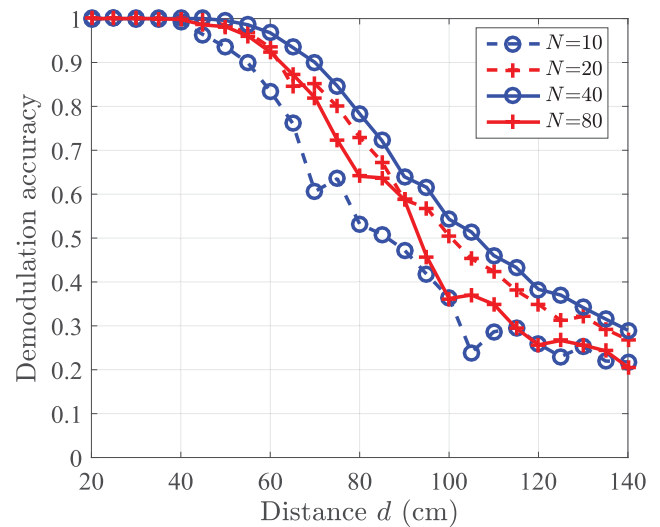

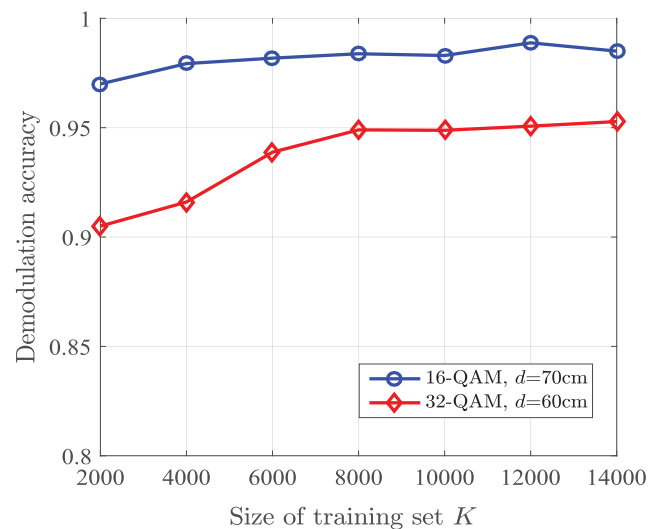**FIGURE 9.** The demodulation accuracy of AdaBoost versus distance *d*.

FIGURE 10. The demodulation accuracy of 16-QAM and 32-QAM modulated signals versus number of training periods *K*.

highest among the five demodulation methods. The accuracies of the DBN and SVM based demodulation methods are similar, but higher than that of both CNN and MLD based demodulation methods. The reason might be referring to the fact that CNNs ignore the classical sampling theorem, so that the performance cannot be guaranteed [37]. Besides, the combined output after down sampling is typically the scalar activity of the most active unit in the pool [38], and the relative position information of parts of waveforms is ignored. Since the practical VLC channels include complex interferences, the MLD classification have a degraded performance.

In Fig. 8 (c) it is shown that for the 256-QAM modulated signals, the demodulation accuracies of all mothods are similar to Fig. 8 (b). Fig. 9 shows the accuracy of AdaBoost based demodulation method versus distance *d* with different
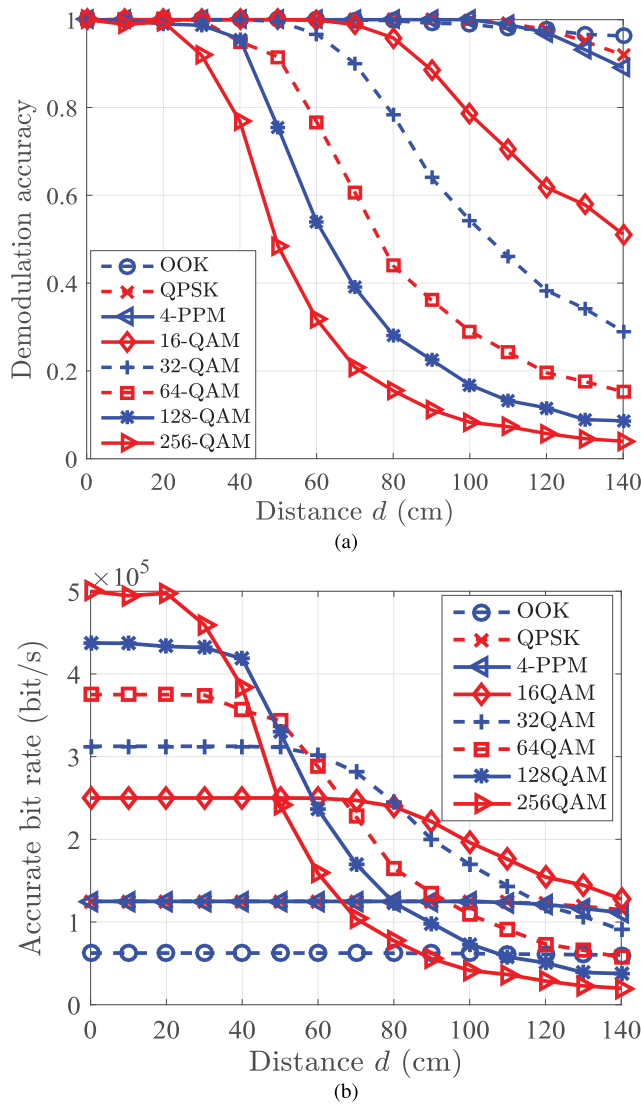
FIGURE 11. (a) The demodulation accuracy of OOK, QPSK, 4-PPM, 32-QAM, 64-QAM, 128-QAM and 256-QAM modulated signals versus distance *d* when *N* = 40. (b) The accurate bit rate of OOK, QPSK, 4-PPM, 32-QAM, 64-QAM, 128-QAM and 256-QAM modulated signals versus distance *d* when *N* = 40.

FIGURE 12. (a) The demodulation accuracy of DBN based method versus epoches when *N* = 40 and *d* = 70cm. (b) The demodulation accuracy of CNN based method versus epoches when *N* = 40 and *d* = 60cm.

numbers of sample points in one period $N = 10, 20, 40, 80$, where the signals are modulated by 32-QAM. The demodulation accuracy increases as number of sample points N increases. Moreover, the demodulation accuracy of the $N = 40$ case is higher than that of $N = 80$ case, while the storage memory of the $N = 40$ case is only a half of that of the $N = 80$ case.

Fig. 10 shows the demodulation accuracy of the AdaBoost demodulation method versus the number of training periods K with 16-QAM modulated signals at $d = 70$cm and 32-QAM modulated signals at $d = 60$cm. For the 16-QAM modulated signal case, the demodulation accuracy increases with the number of training periods K, while when $K \geq 4000$, the demodulation accuracy increases very slowly. Similarly, for the 32-QAM modulated signal case, the demodulation accuracy increases with the number of training periods K,
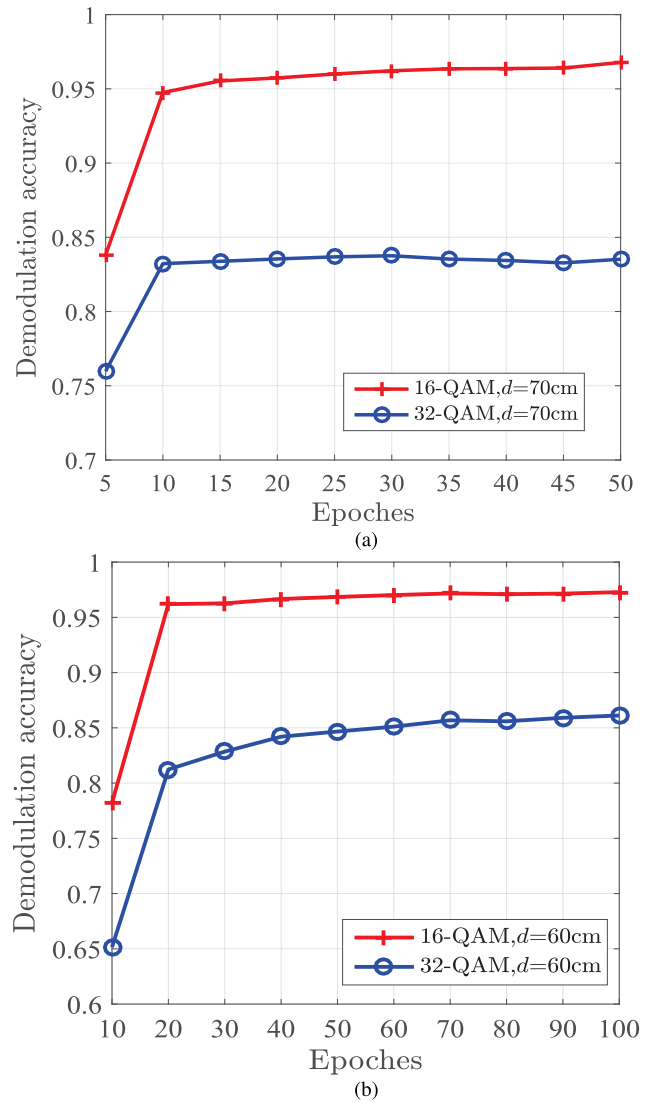
and when $K \geq 8000$, the demodulation accuracy almost keeps the same. Comparing demodulation accuracy of the 16-QAM and 32-QAM modulated signals, it can be observed that more number of training periods K is required for the higher modulation order to achieve a stable accuracy.

Fig. 11 (a) shows the demodulation accuracies of OOK, QPSK, 4-PPM, 32-QAM, 64-QAM, 128-QAM and 256-QAM modulated signals versus distance d when $N = 40$, respectively. We can see that the demodulation accuracies of the eight modulation schemes decrease as the distance d increases. Moreover, for a given distance d, the demodulation accuracies of the eight modulation schemes decrease as the modulation order increases, and the higher the modulation order is, the faster of the rate decreases.

Fig. 11 (b) shows the accurate bit rates[3] of the eight modulation schemes versus distance d. As distance d increases, the effective rates of the eight modulation schemes decrease. When $d \leq 30$cm, the effective rate of 256-QAM is the highest. When 40 cm $< d \leq 50$cm, the highest effective rate is obtained by the 128-QAM modulation scheme. As distance d increases from 50cm to 140cm, the highest accurate bit rate is obtained with 64-QAM, 32-QAM and 16-QAM in turn. Therefore, for short distance or high SNR scenario, high order modulation is preferred.

Fig. 12 (a) shows the demodulation accuracies of the DBN demodulation method versus epochs in BP process with $N = 40$ and $d = 70$cm. We can see that the demodulation accuracies of 16-QAM and 32-QAM modulated signal increase as the number of epoch increases. For the two modulation schemes, the demodulation accuracy increases fast when the number of epoch is less than 10, while the performance of 16-QAM is higher than 32-QAM. When larger than 10, the increasing of epoch number brings limited benefits. Fig. 12 (b) shows the demodulation accuracies of the CNN demodulation method versus epochs when $N = 40$ and $d = 60$cm. The demodulation accuracies of two modulation schemes are similar to Fig. 12 (a).

VII. CONCLUSION

In this paper, three data-driven demodulators (CNN, DBN, and AdaBoost) based demodulators are designed for the physical layer of VLC systems. A flexible end-to-end VLC system prototype is constructed for real data collection. By using the proposed prototype, an open online real modulated dataset is created, which consists eight types of modulated signals, i.e., OOK, QPSK, 4-PPM, 16-QAM, 32-QAM, 64-QAM, 128-QAM and 256-QAM. Based on this real dataset, we investigate the demodulation performance of the proposed three demodulators. Experimental results show that for a given transmission distance, the demodulation accuracy decreases as the modulation order increases. Moreover, that the demodulation accuracy of the AdaBoost based demodulators is higher than other demodulators. For the short distance or high SNR scenario, a high-order modulation is preferred. In the future, we will further investigate dedicated ML based demodulators for VLC systems.

REFERENCES

[1] V. Chandrasekhar, J. G. Andrews, and A. Gatherer, "Femtocell networks: A survey," *IEEE Commun. Mag.*, vol. 46, no. 9, pp. 59–67, Sep. 2008.

[2] *IEEE Standard for Local and Metropolitan Area Networks–Part 15.7: Short-Range Wireless Optical Communication Using Visible Light*, IEEE Standard 802.15.7-2011, Sep. 2011, pp. 1–309.

[3] T. Komine and M. Nakagawa, "Fundamental analysis for visible-light communication system using LED lights," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 100–107, Feb. 2004.

[4] H. Elgala, R. Mesleh, and H. Haas, "Indoor optical wireless communication: Potential and state-of-the-art," *IEEE Commun. Mag.*, vol. 49, no. 9, pp. 56–62, Sep. 2011.

[5] S. Arnon, J. Barry, G. Karagiannidis, R. Schober, and M. Uysal, *Advanced Optical and Wireless Communications Systems*, 1st ed. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[6] A. Jovicic, J. Li, and T. Richardson, "Visible light communication: Opportunities, challenges and the path to market," *IEEE Commun. Mag.*, vol. 51, no. 12, pp. 26–32, Dec. 2013.

[7] P. H. Pathak, X. Feng, P. Hu, and P. Mohapatra, "Visible light communication, networking, and sensing: A survey, potential and challenges," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2047–2077, 4th Quart., 2015.

[8] T. V. Pham, H. Le-Minh, and A. T. Pham, "Multi-user visible light communication broadcast channels with zero-forcing precoding," *IEEE Trans. Commun.*, vol. 65, no. 6, pp. 2509–2521, Jun. 2017.

[9] H. Shen, Y. Deng, W. Xu, and C. Zhao, "Rate-maximized zero-forcing beamforming for VLC multiuser MISO downlinks," *IEEE Photon. J.*, vol. 8, no. 1, Feb. 2016, Art. no. 7901913.

[10] T. Fath and H. Haas, "Performance comparison of MIMO techniques for optical wireless communications in indoor environments," *IEEE Trans. Commun.*, vol. 61, no. 2, pp. 733–742, Feb. 2013.

[11] T. Q. Wang, Y. A. Sekercioglu, and J. Armstrong, "Analysis of an optical wireless receiver using a hemispherical lens with application in MIMO visible light communications," *IEEE J. Lightw. Technol.*, vol. 31, no. 11, pp. 1744–1754, Apr. 12, 2013.

[12] K. Ying, H. Qian, R. J. Baxley, and S. Yao, "Joint optimization of precoder and equalizer in MIMO VLC systems," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 9, pp. 1949–1958, Sep. 2015.

[13] T. Mitchell, B. Buchanan, G. Dejong, T. Dietterich, P. Rosenbloom, and A. Waibel, *Machine Learning*. Beijing, China: China Machine Press, 2003.

[14] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.

[15] H. Huang, W. Xia, J. Xiong, J. Yang, G. Zheng, and X. Zhu, "Unsupervised learning-based fast beamforming design for downlink MIMO," *IEEE Access*, vol. 7, pp. 7599–7605, Dec. 2018.

[16] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018.

[17] M. Önder, A. Akan, and H. Doğan, "Neural network based receiver design for software defined radio over unknown channels," in *Proc. 8th Int. Conf. Elect. Electron. Eng.*, Nov. 2013, pp. 297–300.

[18] M. Zhang, Z. Liu, L. Li, and H. Wang, "Enhanced efficiency BPSK demodulator based on one-dimensional convolutional neural network," *IEEE Access*, vol. 6, pp. 26939–26948, 2018.

[19] X. Lin, R. Liu, W. Hu, Y. Li, X. Zhou, and X. He, "A deep convolutional network demodulator for mixed signals with different modulation types," in *Proc. IEEE 15th Int. Conf. Dependable, Autonomic Secure Comput.*, Nov. 2017, pp. 893–896.

[20] M. Fan and L. Wu, "Demodulator based on deep belief networks in communication system," in *Proc. Int. Conf. Commun., Control, Comput. Electron. Eng.*, Jan. 2017, pp. 1–5.

[21] L. Fang and L. Wu, "Deep learning detection method for signal demodulation in short range multipath channel," in *Proc. IEEE 2nd Int. Conf. Opto-Electron. Inf. Process.*, Jul. 2017, pp. 16–20.

[22] H. Lee, I. Lee, and S. H. Lee, "Deep learning based transceiver design for multi-colored VLC systems," *Opt. Express*, vol. 26, no. 5, pp. 6222–6238, 2018.

[23] H. Lee, I. Lee, T. Q. S. Quek, and S. H. Lee, "Binary signaling design for visible light communication: A deep learning framework," *Opt. Express*, vol. 26, no. 14, pp. 18131–18142, Jul. 2018.

[24] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. 13th Int. Conf. Mach. Learn.*, Jul. 1996, pp. 148–156.

[25] J. Sola and J. Sevilla, "Importance of input data normalization for the application of neural networks to complex industrial problems," *IEEE Trans. Nucl. Sci.*, vol. 44, no. 3, pp. 1464–1468, Jun. 1997.

[26] S. Peng *et al.*, "Modulation classification based on signal constellation diagrams and deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 718–727, Mar. 2019.

[27] J. Pan, Y. Yin, J. Xiong, W. Luo, G. Gui, and H. Sari, "Deep learning-based unmanned surveillance systems for observing water levels," *IEEE Access*, vol. 6, no. 1, pp. 73561–73571, 2018.

[28] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2007.

---

[3]The accurate bit rate is the product of demodulate accuracy and the information each symbol carries.

[29] J. Bouvrie, "Notes on convolutional neural networks," Center Biol. Comput. Learn., Massachusetts Inst. Technol, Cambridge, MA, USA, MIT CBCL Tech Rep., 2006, pp. 38–44.

[30] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[31] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, 2002.

[32] L. J. Buturovic and L. T. Citkusev, "Back propagation and forward propagation," in *Proc. Int. Joint Conf. Neural Netw.*, vol. 4, Jun. 1992, pp. 486–491.

[33] I. Mukherjee, C. Rudin, and R. E. Schapire, "The rate of convergence of AdaBoost," *J. Mach. Learn. Res.*, vol. 14, no. 3, pp. 2315–2347, 2011.

[34] G. Rätsch, T. Onoda, and K.-R. Müller, "Soft margins for AdaBoost," *Mach. Learn.*, vol. 42, no. 3, pp. 287–320, 2001.

[35] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[36] R. B. Palm, "Prediction as a candidate for learning deep hierarchical models of data," M.S. thesis, Tech. Univ. Denmark, Lyngby, Denmark, DTU Informatics, E-mail: reception@imm.dtu.dk, 2012. [Online]. Available: http://www.imm.dtu.dk/English.aspx

[37] A. Azulay and Y. Weiss. (2018). "Why do deep convolutional networks generalize so poorly to small image transformations?" [Online]. Available: https://arxiv.org/abs/1805.12177

[38] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature Neurosci.*, vol. 2, no. 11, pp. 1019–1025, Nov. 1999.
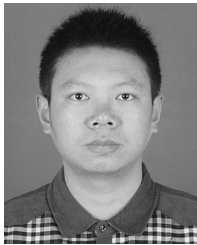
SHUAI MA received the B.S. and Ph.D. degrees in communication and information systems from Xidian University, Xi'an, China, in 2009 and 2016, respectively. From 2014 to 2015, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA. Since 2019, he has been an Associate Professor with the School of Information and Control Engineering, China University of Mining and Technology. He is also with the State Key Laboratory of Integrated Services Networks, Xidian University. His research interests include visible light communication, machine learning, wireless communications, and network information theory.

JIAHUI DAI received the B.S. degree from the China University of Mining and Technology, Xuzhou, China, in 2016, where she is currently pursuing the M.E. degree in information and communication engineering. Her research interests include visible light communication, wireless communications, and machine learning.

SONGTAO LU (S'10–M'18) received the Ph.D. degree in electrical and computer engineering from Iowa State University, in 2018. He is currently a Postdoctoral Associate with the Department of Electrical and Computer Engineering, University of Minnesota Twin Cities, Minneapolis. His primary research interests include wireless communications, optimization, and machine learning. He was a recipient of the Graduate and Professional Student Senate Research Award from Iowa State University, in 2015, the Research Excellence Award from the Graduate College of Iowa State University, in 2017, and the Student Travel Award from the 20th International Conference on Artificial Intelligence and Statistics, in 2017.
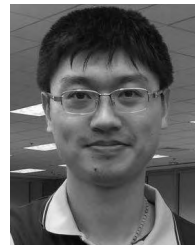
HANG LI (S'13–M'16) received the B.E. and M.S. degrees from Beihang University, Beijing, China, in 2008 and 2011, respectively, and the Ph.D. degree from Texas A&M University, College Station, TX, USA, in 2016. He was a Postdoctoral Research Associate with Texas A&M University, from 2016 to 2017, and the University of California at Davis, from 2017 to 2018. Since 2018, he has been a Visiting Research Scholar with the Shenzhen Research Institute of Big Data, Shenzhen, China. His current research interests include wireless networks, stochastic optimization, and applications of machine learning.

HAN ZHANG received the B.E. degree from the Beijing University of Posts and Telecommunication, Beijing, China, in 2010. He is currently pursuing the Ph.D. degree with the University of California at Davis, USA. His current research interests include wireless networks and machine learning.

CHUN DU received the B.S. degree from the China University of Mining and Technology, Xuzhou, China, in 2011, and the M.E. degree from Duisburg-Essen University, Westfalen, Germany, in 2014. He is currently pursuing the Ph.D. degree in information and communication engineering with the School of Information and Control Engineering, China University of Mining and Technology. His research interests include visible light communication, wireless communications, and physical layer security.

SHIYIN LI received the Ph.D. degree in information and communication engineering from the China University of Mining and Technology, Xuzhou, China, in 2010. Since 2010, he has been a Professor with the School of Information and Control Engineering, China University of Mining and Technology, where he is currently the Head of the Department of Information Engineering. His research interests include wireless communications and network congestion control.

• • •