Equalizing QAM Waveform Distortion with Linear SVM Classifier and its Machine Learning Dataset Generation

Yiwei Liu†, Yukina Haruta‡, Yutaka Masuda†, Tohru Ishihara†
†Graduate School of Informatics, Nagoya University, Furo-cho, Chikusa-ku, Nagoya 464-8601 Japan
‡School of Informatics, Nagoya University, Furo-cho, Chikusa-ku, Nagoya 464-8601 Japan

Abstract— Multi-level modulation formats such as quadrature amplitude modulation (QAM) have been widely used in modern wideband communications. QAM waveforms are distorted by various influences during communication. To learn the distortion trends in the QAM waveform accurately, this paper first focuses on generating datasets for QAM communications. The paper then proposes a method for equalizing QAM waveform distortion with a support vector machine (SVM) classifier which identifies the received QAM code based on pre-learned distortion trends. Simulation results obtained for our SVM classifier designed as a dedicated digital circuit demonstrate that the proposed method reduces the computational cost by 80% compared to existing methods that achieve the similar classification accuracy. We also confirmed that the classification accuracy was improved from 96.4% to 99.0% at 1.4 times the computational cost compared to one of the simplest existing SVM classifiers.

I. Introduction

With the advent of 5G and 6G, telecommunications networks must support a wide range of emerging services, such as extended reality, digital twin, massive communication, and fixed wireless access (FWA), while ensuring complex quality-of-service (QoS) requirements. As these communications services become more widespread, network traffic is growing rapidly, making it necessary to further expand the capacity of networks in a sustainable and economical way. To meet the growing demand for high-capacity data communications, higher-order quadrature amplitude modulation (QAM) formats have attracted widespread attention. However, high-order QAM makes the constellation much denser, which makes it very sensitive to the nonlinear distortion. Thus, the correct detection of high-order QAM at the receiver side is usually very difficult due to the nonlinear distortion. Recently, digital signal processing based on machine learning (ML) has been extensively studied for equalization and demodulation of broadband communications systems [1]. Despite its potential, progress in ML technologies in the field of network communications has been slowed by limited availability of high-quality, publicly accessible datasets [2].

With this background, this paper first focuses on generating datasets for ML applications such as ML-based equalization of QAM symbols. QAM is a multi-level modulation format that encodes information using both amplitude and phase. It combines two orthogonal carrier waves that are modulated independently in amplitude to transmit data efficiently. For example, 16-QAM encodes 4 bits per symbol, representing 16 discrete values (i.e., 0000_2 to 1111_2). To accurately understand the waveform distortion in QAM communications, we design a QAM transceiver model and use it to generate datasets for QAM communications using a commercial optoelectronic circuit simulator. In the dataset generation, we consider phase and amplitude modulation noises with the model to reflect the actual situation of QAM communications.

This paper next proposes a method to equalize QAM waveform distortion with a support vector machine (SVM) classifier. SVM is one of the most studied supervised ML models. It is based on the max-margin algorithm that maximizes the margin of the two classes in the training dataset. SVMs are broadly divided into linear SVMs and nonlinear SVMs. Nonlinear SVMs need to calculate the dot product of all support vectors and the input during classification. Therefore, they need a large amount of calculations while improving classification accuracy [3]. In this paper, prioritizing area and energy efficiency, we propose a method based on the linear SVM. It uses the boundary created by concatenating line segments on the I-Q plane to classify each bit value of the received QAM symbol into two classes (i.e., 1 or 0). HDL-based simulation results obtained for our SVM classifier designed as a dedicated digital circuit demonstrate that the proposed method reduces the computational cost by 80% compared to existing methods that achieve the similar classification accuracy. We also confirmed that the classification accuracy is improved from 96.4% to 99.0% at the cost of a 40% increase in the number of MAC operations compared to one of the simplest existing SVM classifiers.

The remainder of the paper is organized as follows; Section II summarizes previous works related to equalization of QAM waveform distortion and generation of machine learning datasets for QAM communications. Section III presents the details of our approach to generating machine learning datasets that help improve the accuracy of QAM

equalization. Section IV proposes a linear SVM-based method for decoding QAM symbols. Section V shows the experimental results obtained using a HDL-based hardware simulator. Section VI concludes this paper.

II. Related Work

A. Realistic Dataset Generation for Wireless Modulation Classification

In [4], the authors propose RML22, a benchmark dataset designed to address limitations in previous datasets such as RML16. By correcting issues related to channel modeling, artifact application order, and signal parameterization, RML22 achieves a 23% accuracy improvement in modulation classification tasks. The study adopts a data-centric approach, emphasizing the importance of realistic signal simulation and systematically analyzing the effects of channel, clock, and noise artifacts on classification performance.

B. Voting-Based Deep Convolutional Neural Networks (VB-DCNNs) for M-QAM and M-PSK Signals Classification

In [5], a deep learning-based automatic modulation classification (AMC) approach is proposed using a voting-based DCNN (VB-DCNN) to identify M-QAM and M-PSK signals. By simulating modulated waveforms and transmitting them through fading channels with additive white Gaussian noise (AWGN), a large and diverse dataset is generated. The dataset is then split into training, validation, and testing subsets to train multiple CNN instances. Notably, the paper highlights that high classification accuracy (up to 99.7% for 16-QAM) is achievable even under noise, showcasing the effectiveness of machine learning models when supported by rich and well-structured datasets. This underscores the critical role of synthetic waveform generation in advancing data-driven equalization and classification systems.

C. Modulation Recognition Based on Constellation Diagram for M-QAM Signals

In [6], the authors propose a non-cooperative modulation recognition method for M-QAM signals based on constellation diagram clustering. By estimating parameters such as carrier frequency and baud rate directly from the received signals and applying K-means clustering to reconstruct the constellation diagram, the method achieves high recognition accuracy without prior information. Crucially, this work demonstrates the feasibility of extracting modulation features from noisy constellations to classify signal types like 16-QAM, 32-QAM, and 64-QAM, underscoring the value of clean and structured datasets for training machine learning models in signal classification and equalization tasks.

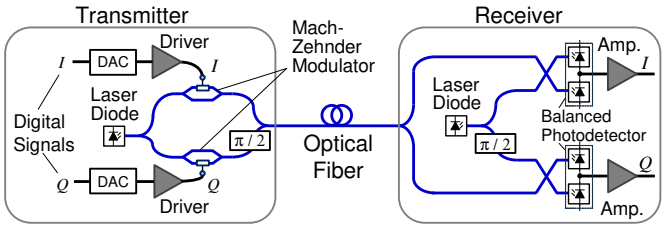


Fig. 1. Optical transceiver circuit used in optoelectronic circuit simulator for machine learning dataset generation.

III. Machine Learning Dataset Generation

A. QAM Optical Transceiver

Our approach to generating QAM datasets for machine learning applications is based on circuit simulation of QAM optical transceivers. We use a commercial optoelectronic circuit simulator for the dataset generation. Figure 1 shows an example of the optical QAM transceiver circuit used in the simulation. DAC in Fig. 1 represents a digital-to-analog converter. It converts the digital I-Q inputs into analog signals to control the Mach-Zehnder Modulator which modulates the optical signal generated by the laser diode in the transmitter circuit. The upper and lower Mach-Zehnder modulators generate optical modulation signals independently. After the modulation, the upper signal remains in-phase (I), and the lower signal shifts its phase by $\pi/2$, or 90 degrees (Q). These two optical signals are then combined by an optical power combiner and sent to the optical fiber.

The receiver circuit shown in Fig. 1 is based on a typical homodyne receiver structure. The optical signal incoming from the optical fiber is first divided into the two signals. The local optical signal generated by the laser diode in the receiver side is also divided in two. One of the divided local signals is phase shifted by $\pi/2$. The locally generated two signals are mixed with the two divided incoming signals, respectively. The mixed signals are then converted into electrical signals (i.e., I and Q) by the balanced photo-detectors. The balanced photo-detectors subtract the converted photo-currents from each other, resulting in the cancellation of common mode noise.

B. Noise and Interference in QAM Transceivers

The I and Q signals generated in the transmitter circuit are inherently correlated since these signals are generated from the same laser source. If there is noise in the laser source, it will be copied to both the optical I and Q signals. Since the same driver circuit shown in Fig. 1 is used to control the modulator, there is a temporal correlation between successive signal waveforms in the transmitter. Those spatiotemporal correlations cause QAM waveform distortions in the transmitter side. Noises generated in the driver circuits are converted into amplitude noises and phase noises of the optical signals by the

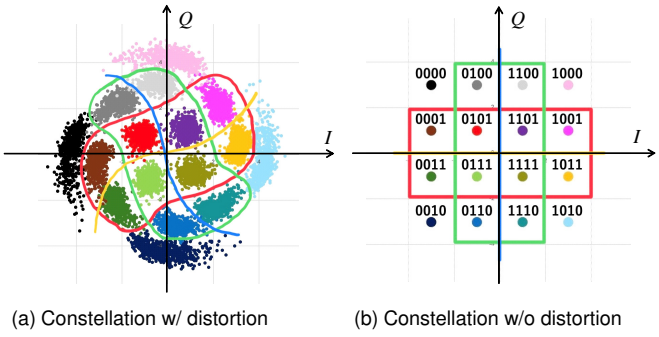


Fig. 2. QAM constellation diagram with and without distortion.

Mach-Zehnder modulators. These noises also cause distortion of the transmitting waveforms.

Similar to the transmitter, the two amplifiers corresponding to the I and Q outputs in the receiver have a spatial correlation from each other since the optical signals are incoming from the same laser diode. There is also a temporal correlation between successive electrical waveforms generated by the same amplifier circuit. Those correlations and noises cause the distortion of the waveforms received. To reflect all those noises and correlations into the datasets, we use an optoelectronic circuit simulator and the transceiver model shown in Fig. 1 when generating the machine learning datasets.

IV. SVM Classification for QAM Equalization

A. *Bit-wise classification of QAM symbols*

SVM-based methods that learn QAM waveform distortion trends from a large amount of communications data and compensate the distortion based on these pre-learned trends have been proposed in [7–10]. An example of a 16-QAM constellation is shown in Fig. 2. The received symbol of 16-QAM is encoded and decoded as a 4-bit code since 16 coordinate points in the I-Q plane can be represented by a 4-bit code. In the method proposed in [7], the 0-1 classification for each bit in the code representing the received symbol is simplified to a 2-class classification problem using SVM. For example, the least significant bit (LSB) of a 4-bit code is classified as 1 if the I-Q coordinate of the received symbol is within the red frame in Fig. 2(a), and as 0 if it is outside the red frame. Similarly, the most significant bit (MSB) is classified as 1 if the received symbol is above and to the right of the blue line in Fig. 2(a), and as 0 if it is below and to the left.

B. *Basic idea of proposed approach*

Our approach is based on the idea presented in [7]. However, unlike the methods proposed in [7–10], our method is accurate and less expensive in computational cost. Those existing methods [7–10] are either based on linear SVM, which is low cost but has low classification accuracy, or based on nonlinear SVM, which is highly accurate but expensive. Our method classifies each bit of a received symbol into two classes (i.e., 1 or 0) using boundaries expressed by a concatenation of linear equations in the I-Q plane, which are derived by simple linear SVMs. For example, the constellation map shown in the upper left of Fig. 3 shows an example of decoding the LSB in the received 16-QAM symbol by classifying the inside and outside of the area enclosed by the thick blue boundary. This boundary is composed of the eight lines derived from the linear SVMs. In summary, the problem of decoding each bit of a received 16-QAM symbol can be defined as the problem of dividing the I-Q plane into a region of 0 or 1 with a boundary represented with a concatenation of multiple lines which are derived by linear SVMs. In the case of 16-QAM, our method uses 8 lines for the least significant bit (LSB) and the third LSB, and 3 lines for the second LSB and the most significant bit (MSB) as shown in Fig. 3. In other words, the received 16-QAM symbol can be decoded into 4 bits by dividing the I-Q plane with boundaries represented by a concatenation of lines which can be derived with a total of 22 linear SVMs.

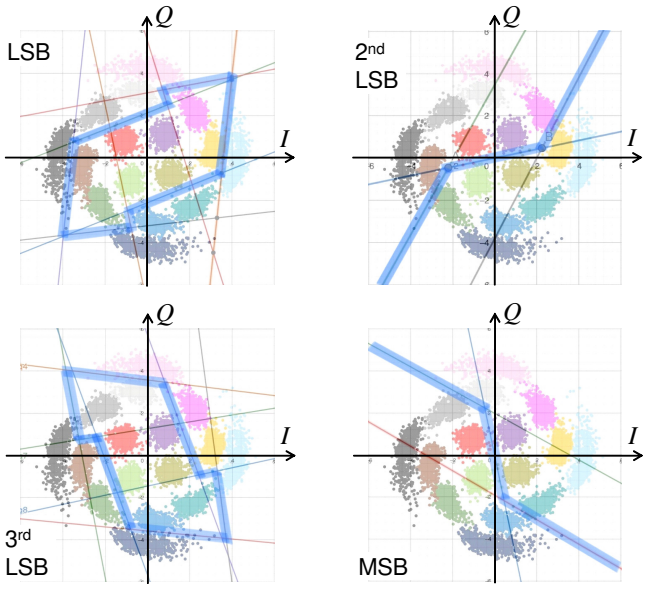


Fig. 3. Equalizing QAM waveform distortion with line segment connections derived by linear SVM classifier.

C. *Hardware implementation of SVM classifier*

Any line in the I-Q plane can be expressed by a linear equation (1) with I and Q as variables. If the value of E_x is 0, the I-Q coordinate of the received symbol in the I-Q plane is on the line.

$$E_x = \alpha \cdot I + \beta \cdot Q + \gamma \tag{1}$$

Therefore, the side of a particular line in the I-Q plane on which the coordinate of a received QAM symbol is located can be determined by the sign bit (i.e., the most

significant bit) of the result of a multiply-add operation which consists of two multiplications and two additions. Figure 4 shows the hardware implementation for the LSB. The received I-Q signals are first converted from analog signals to digital signals using AD converters (ADCs in Fig. 4). The digital I-Q values are multiplied with the α and β values respectively, and these products are finally added together with γ as shown in Fig. 4. Note that the symbols "$*$" and "$+$" in Fig. 4 represent digital multiplier and adder, respectively. The values of α, β and γ are obtained by a linear SVM classifier. Once the value of E_x in equation (1) is calculated with the digital multiply-add circuits, the sign bits of those results are obtained. The inverts of the sign bits are used to calculate each bit of the code received. Figure 5 shows an example of the logic circuit used to calculate the second LSB of the received code. The AND gate in Fig. 5 specifies the green shaded area of the I-Q constellation diagram. Similarly, the OR gate covers either the green shaded area or the yellow shaded area of the I-Q constellation diagram, which corresponds to the second LSB of the received code. This idea can be generalized for the other bits of the code received.

V. Experimental Evaluation

A. Accuracy, Power, and Area Estimation

A.1. Evaluation Setup

As benchmark datasets, we use two types of datasets to evaluate the proposed SVM classifier. The first dataset is synthetically created by incorporating noise and spatiotemporal correlations based on numerical manipulation without using a circuit simulator. The second dataset is generated by injecting amplitude and phase noise into the transceiver circuit model shown in Fig. 1 using an optoelectronic circuit simulator. In generating the second dataset, we inject random noise into both the transmitter and receiver laser diodes. Random noise is also injected for the driver circuits in the transmitter and the amplifier circuits in the receiver. Figure 6 shows examples of constellation maps corresponding to the first and second datasets. For both datasets, 90% of the data is used for training and 10% is used for evaluation. To evaluate the proposed SVM classifier, we compare it with the following three existing SVM classification methods. The results of those methods are compared in Table I and II.

OvO linear SVM is to generate an optimal linear boundary between every two different classes (i.e., One versus One; OvO) of training datasets. The advantage of OvO SVM is that the number of samples per training is relatively small. Therefore the training speed for finding a single boundary is faster and the accuracy is higher. However, since the N classification problem needs to train $N \times (N-1)/2$ decision functions ($N > 2$), the total number of linear boundaries will be too large when N is large and thus the prediction speed will be affected.

OvR linear SVM is to generate a boundary between a class of samples and the remaining multiclass samples (i.e., One versus Remain; OvR), to achieve multiclass recognition. This method only requires an optimal boundary between a class of samples and the corresponding remaining samples, rather than classifying between the two samples. Therefore, if it is a N classification problem, then you need N linear SVMs ($N > 2$) and the prediction speed is faster. However, the classification accuracy is generally lower than that of the OvO.

RBF nonlinear SVM is based on the Radial Basis Function (RBF) kernel, which is one of the most powerful, useful, and popular kernels in the SVM classifiers. However, it needs to calculate the dot products of all support vectors and the input during classification. Therefore, the computational complexity depends on the number of support vectors, and the number of MAC operations needed is generally much higher than the OvO linear SVM.

We implement four SVM classifiers including our proposed SVM classifier and above three existing methods using the scikit-learn framework. Scikit-learn is a Python module for machine learning.

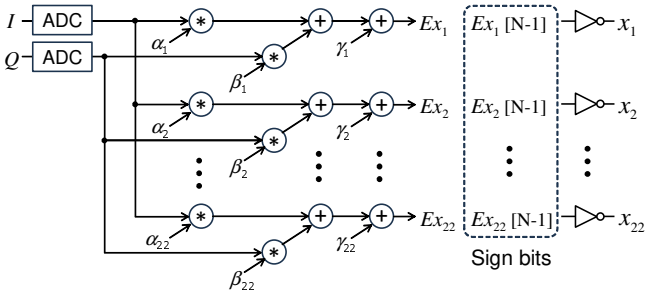


Fig. 4. Logic circuit implementation of linear SVM classifier.

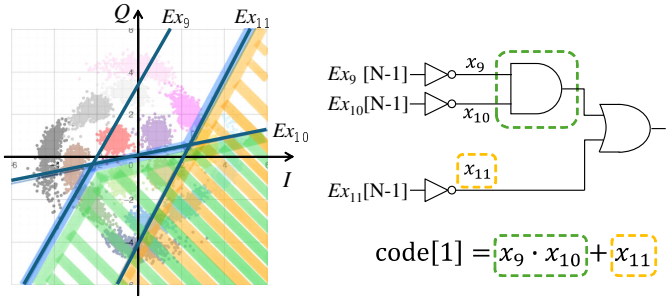
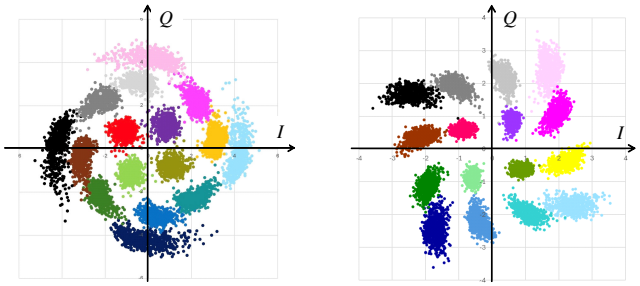


Fig. 5. Example of logic circuit implementation for 2^{nd} LSB of received 4-bit code.



(a) Constellation of the 1st dataset (b) Constellation of the 2nd dataset

Fig. 6. Examples of constellation diagrams corresponding to the first and second datasets.

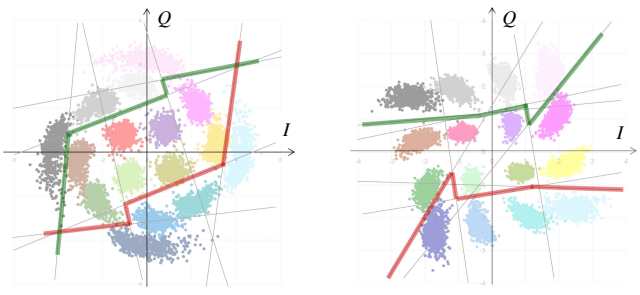
TABLE I
CLASSIFICATION ACCURACY FOR THE 1^{st} DATASET.

SVM model	accuracy (%)	# MAC
OvO linear SVM	99.04	240
OvR linear SVM	96.39	32
RBF nonlinear SVM	99.26	> 10,000
Proposed SVM	99.01	44

A.2. Evaluation Results

We conduct experiments using the datasets presented in the previous subsection. 10,000 communication symbols are used for the accuracy evaluation. The results for the two datasets are summarized in Table I and II, respectively. The three existing methods described in the previous subsection are compared with our proposed method in the tables. The word "# MAC" represents the number of multiply-accumulate (MAC) operations required for the SVM classification. Note that two MAC operations are needed to express one linear equation with I and Q as variables on the I-Q plane as explained in Sec. IV.

Compared to OvO linear SVM, our method reduces the number of MAC operations needed by more than 80% while achieving similar classification accuracy. Compared to OvR, which is one of the most lightweight SVM classifiers, classification accuracy is improved from 96.4% to more than 99.0% at the cost of only 40% increase in the number of MAC operations needed. Although the classification accuracy achieved by RBF nonlinear SVM is very good, the number of MAC operations required is huge.



(a) Constellation of the 1st dataset (b) Constellation of the 2nd dataset

Fig. 7. Examples of SVM classification for the LSBs in constellations corresponding to the first and second datasets.

TABLE II
CLASSIFICATION ACCURACY FOR THE 2^{nd} DATASET.

SVM model	accuracy (%)	# MAC
OvO linear SVM	99.74	240
OvR linear SVM	96.42	32
RBF nonlinear SVM	99.71	> 10,000
Proposed SVM	99.51	44

B. *Generalization of the proposed method*

Our SVM classification method can be generalized as a method to compensate for various 16-QAM constellation distortions which correspond to distortions in the received QAM waveform. For example, Figure 7 shows how the least significant bit (LSB) of the received code can be decoded taking into account the variation of the distortion in the 16-QAM constellation. The figures on the left and right show constellations for the 1^{st} and 2^{nd} datasets, respectively. The decoding process is achieved by identifying the region bounded by two zigzag red and green borders. Any region bounded by two zigzag borders composed of four line segments, respectively, can be specified using eight linear equations and AND-OR logic functions, as explained in Fig. 5. However, the combination of AND-OR logic changes depending on the distortion of the constellation. We utilize a multiplexer-based circuit shown in Fig. 8 to make the AND-OR combinations programmable. For example, the region between the two zigzag boundaries in the left constellation of Fig. 7 can be specified by $((x_1 \wedge x_2) \vee (x_3 \wedge x_4)) \wedge ((x_5 \wedge x_6) \vee (x_7 \wedge x_8))$. On the other hand, the region between the two zigzag boundaries in the right constellation is specified by $((x_1 \vee x_2) \wedge (x_3 \vee x_4)) \wedge ((x_5 \vee x_6) \wedge (x_7 \vee x_8))$. With this programmable logic, both regions can be specified. The circuit shown in Fig. 8 is just an example of a naive circuit realization; more sophisticated circuit implementations may exist. The regions corresponding to the other

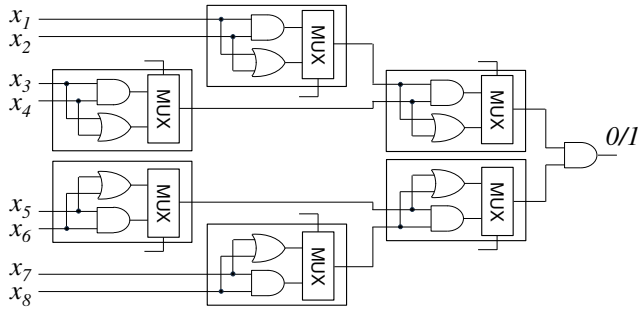


Fig. 8. Examples of a programmable logic function.

bits of the code received can be specified similarly.

We describe a register transfer level (RTL) Verilog-HDL description of the SVM classifier which decodes the 16-QAM communication symbol. It has a functionality to correctly decode the received waveform taking into account the variation of the distortion in the 16-QAM constellation as explained above. Based on the RTL simulation, we confirm that the classification accuracy on both the 1^{st} and 2^{nd} datasets is exactly the same as the results obtained with Python-code-based accuracy estimation.

VI. Conclusion

In this paper, we proposed a method to equalize QAM waveform distortion with a support vector machine (SVM) classifier. The proposed SVM classifier uses the boundary created by concatenating line segments on the I-Q plane to classify each bit value of the received QAM symbol into two classes (i.e., 1 or 0). We use two types of datasets to evaluate the proposed SVM classifier. The first dataset is synthetically created by incorporating noise and spatiotemporal correlations based on numerical manipulation without using a circuit simulator. The second dataset is generated by injecting amplitude and phase noise into the transceiver circuit model shown in Fig. 1 using an optoelectronic circuit simulator. We designed the proposed SVM classifier using Verilog-HDL. With the HDL description, we confirmed that it could successfully equalize 16-QAM communication symbols and decode them into the correct 4-bit digital signal with high classification accuracy for both datasets. Compared to OvO linear SVM, our method reduces the number of MAC operations needed by more than 80% while achieving similar classification accuracy. Compared to OvR, which is one of the simplest existing SVM classifiers, classification accuracy is improved from 96.4% to more than 99.0% at the cost of only 40% increase in the number of MAC operations needed. Although the classification accuracy achieved by RBF nonlinear SVM is very good, the number of MAC operations required is huge. Our method involves much fewer MAC operations and incurs only a 0.2% accuracy degradation.

Our future work will be devoted to fully automate the boundary generation for SVM classification. Generating more different datasets taking practical noise and interference into account to enhance the classification accuracy of our SVM classifier is also our important future work.

Acknowledgement

This work is partly supported by MEXT/JSPS KAKENHI Grant Number 24H00072, JST CRONOS Grant Number JPMJCS24K1 and JST CREST Grant Number JPMJCR21C3.

References

[1] M. Mehta, "An AI Compute ASIC with Optical Attach to Enable Next Generation Scale-Up Architectures," in *2024 IEEE Hot Chips 36 Symposium (HCS)*, August 2024, pp. 1–30.

[2] C. Fischione, M. Chafii, Y. Deng, and M. Erol-Kantarci, "Data Sets For Machine Learning In Wireless Communications And Networks," *IEEE Communications Magazine*, vol. 61, no. 9, pp. 80–81, 2023.

[3] M. H. Yacoub, S. M. Ismail, L. A. Said, A. H. Madian, and A. G. Radwan, "Support Vector Machine reconfigurable hardware implementation on FPGA," *Franklin Open*, vol. 7, p. 100115, 2024.

[4] V. Sathyanarayanan, P. Gerstoft, and A. E. Gamal, "RML22: Realistic Dataset Generation for Wireless Modulation Classification," *IEEE Transactions on Wireless Communications*, vol. 22, no. 11, pp. 7663–7675, 2023.

[5] M. Talha, M. Sarfraz, A. Rahman, S. A. Ghauri, R. M. Mohammad, G. Krishnasamy, and M. Alkharraa, "Voting-Based Deep Convolutional Neural Networks (VB-DCNNs) for M-QAM and M-PSK Signals Classification," *Electronics*, vol. 12, no. 8, 2023.

[6] C. Zhendong, J. Weining, X. Changbo, and L. Min, "Modulation recognition based on constellation diagram for M-QAM signals," in *2013 IEEE 11th International Conference on Electronic Measurement Instruments*, vol. 1, 2013, pp. 70–74.

[7] W. Chen, J. Zhang, M. Gao, and G. Shen, "Performance improvement of 64-QAM coherent optical communication system by optimizing symbol decision boundary based on support vector machine," *Optics Communications*, vol. 410, pp. 1–7, 2018.

[8] C. Wang, J. Du, G. Chen, H. Wang, L. Sun, K. Xu, B. Liu, and Z. He, "QAM classification methods by SVM machine learning for improved optical interconnection," *Optics Communications*, vol. 444, pp. 1–8, 2019.

[9] Y. Sato, T. Kyono, K. Ikuta, Y. Kurokawa, and M. Nakamura, "SVM-based non-linearity equalization for optical communication systems," in *2020 International Conference on Emerging Technologies for Communications (ICETC2020)*, December 2020, pp. I1–1.

[10] ——, "Equalization of optical nonlinear waveform distortion using SVM-based digital signal processing," *IEICE Communications Express*, vol. 10, no. 8, pp. 552–557, 2021.