IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 17 June 2024, accepted 12 August 2024, date of publication 16 August 2024, date of current version 28 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3444796

RESEARCH ARTICLE

Identification of Leaf Disease Based on Memristor Convolutional Neural Networks

NENGYUAN PAN, WEIMING YANG, YUTING LUO, AND YONGLIN WANG
Faculty of Artificial Intelligence, Hubei University, Wuhan 430062, China

Corresponding author: Weiming Yang (20040416@hubu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61601175.

ABSTRACT Deep learning methods based on convolutional neural networks can identify subtle disease features in plant leaves, thereby improving the accuracy and efficiency of plant leaf disease detection. Traditional convolutional neural network models have more parameters, lower training efficiency, and require a large amount of computing resources. A TTN-MobileNetV2 neural network model based on memristors for plant leaf disease detection is proposed in this paper. Firstly, integrate the Triplet attention module into the backbone structure of the network to capture local features, and utilize Cross-Norm and Self-Norm(CNSN) normalization techniques to enhance the generalization robustness under distribution changes. In addition, a Mish activation function with enhanced nonlinear characteristics was introduced to improve the accuracy of neural network detection. Experimental results on the Plant Village and Rice leaf disease datasets showed identification accuracies of 99.03% and 99.16%, respectively. On this basis, using the MemTorch simulation environment, the weights of all convolutional layers and fully connected layers in the convolutional neural network are mapped to the conductance values of the memristors in the cross array of memristors, completing the implementation of the memristor TTN-MobileNetV2 network. The performance of the memristor network was tested using two types of memristor models: linear ion drift model and data-driven Verilog-A RRAM. The recognition accuracy losses of the TTN-MobileNetV2 memristor network corresponding to the two memristor models were 0.32, 0.34, and 0.52, 0.61, respectively. So the memristor convolutional neural network can meet the performance requirements of plant leaf disease recognition and has inherent advantages of high speed and low power consumption.

INDEX TERMS Memristor, convolutional neural network, identification of leaf disease, MobileNetV2.

I. INTRODUCTION

Leaf disease identification is essential in modern agriculture, improving productivity while ensuring food safety and environmental protection. Early identification and diagnosis of leaf diseases allow farmers to take effective control measures, enhancing yield and quality. Convolutional neural networks (CNNs) are widely used for leaf disease detection due to their strong performance in image analysis and pattern recognition [1]. CNNs can extract complex disease features from numerous plant leaf images, enabling early recognition and classification of leaf diseases, which helps prevent disease spread and crop yield loss [2]. For example, Lee et al. evaluated several popular convolutional neural network models on the Plant Village (PV) dataset [3]. The training accuracy of the convolutional networks was generally above 90%, confirming that the application of convolutional neural networks is very effective for the classification of crop leaf diseases. However, deep neural networks often have a large number of parameters, leading to longer inference times and thus affecting the performance of real-time applications on edge devices. Sethy et al. constructed a dataset for rice leaf disease (RLD) and combined the MobileNetV2 network model with a support vector machine model, achieving a recognition accuracy of 97.96% on this dataset [4]. Although this idea is great, introducing SVM for large classification datasets may significantly extend training speed.

However, due to limitations in computing speed and power consumption, deploying CNN models on intelligent terminals requires lightweight design and embedded system

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

implementation. This can reduce system performance to some extent. Additionally, the separation of processors and memory in von Neumann-based embedded systems leads to the von Neumann bottleneck.

Neuromorphic architecture, a biomimetic computing approach, is a promising solution to the issues mentioned. In a neuromorphic system, neurons (simple computing units) and synapses (local memory units) are interconnected. The structure helps to overcome the transmission bottleneck. Memristors, with their ideal memory functions, can adjust resistance by controlling input magnetic flux or charge. Memristors also have advantages such as nanoscale [5], low power consumption [6], nanosecond switching speed [7], and long durability. Studies have proven that memristor-based neural networks have significant advantages in energy efficiency over traditional von Neumann architectures [8], [9].Implementing memristors as weights for neural networks has become an excellent candidate solution for neural morphology synapses. The cross array based on memristors adopts a non von Neumann computing paradigm and a storage computing integrated architecture, eliminating the problem of data relocation between logic processors and storage chips, reducing energy consumption and latency [10], and is considered an ideal choice for hardware implementation of neural morphology computing [11]. In 2020, a research team from Tsinghua University constructed a five-layer memristor convolutional neural network (mCNN) and successfully completed image recognition tasks on the MNIST dataset [12]. In 2023, the team developed a fully integrated, neuron inspired memristor chip with on-chip learning capability and low energy consumption [13]. So, combining memristors with deep learning is a very promising approach for intelligent terminal deployment.

In this paper, a lightweight TTN-Mobilenetv2 network with an attention mechanism is proposed to classify the PV and the RLD datasets. The TTN-MobileNetV2 network based on memristors with a reverse residual structure is constructed, then the constructed memristor neural network is used to the leaf disease detection. The experiment results indicate that the memristors TTN-MobileNetV2 which consumes less energy compared to neural networks implemented with conventional von Neumann architecture hardware, can achieve an advanced level on the PV and RLD datasets.

II. IMPROVED CONVOLUTIONAL NEURAL NETWORK

A. MobileNetV2

MobileNetV2 incorporates inverted residual structures to mitigate gradient vanishing issues while enriching feature representation in deep networks [14]. Moreover, the convolutional layers of the MobileNetV2 network utilize lightweight deepwise separable convolutions, which initially process spatial features through individual convolutional kernels for each input channel, followed by 1 × 1 pointwise convolutions to combine features from different channels. The inverted residual structures optimize both spatial features and the depth of the kernels, significantly reducing the number of parameters and computational load. The structure of the MobileNetV2 network is shown in FIGURE 1.

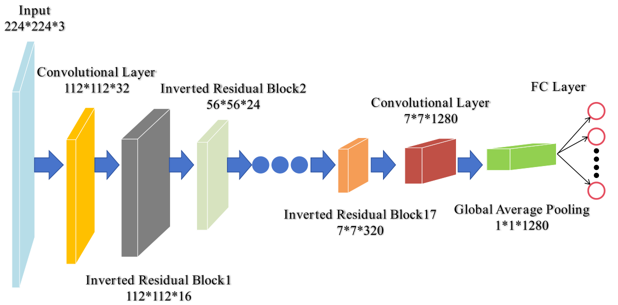
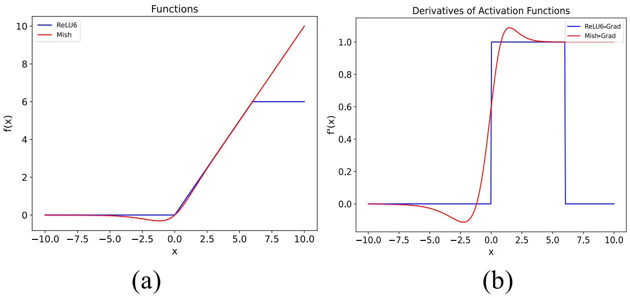


FIGURE 1. MobileNetV2 network structure.

B. IMPROVED MobileNetV2

This research enhances the MobileNetV2 architecture, focusing on modifications within the Inverted Residual Blocks. We implemented three key changes. First, we replaced the ReLU6 activation function with the smoother Mish function. As shown in Fig. 2(a), ReLU6 sets negative inputs to zero and truncates values above 6, potentially losing negative information. In contrast, the Mish functions offer an adaptive output range (including negative outputs), enabling more effective utilization of input information while modulating the degree of non-linearity in activations. The derivative plots of two activation functions, as shown in Fig. 2(b), indicate that the derivative of ReLU6 is consistently one between zero and six and zero elsewhere which can lead to the loss of training information when inputs fall within the zero-output region. Conversely, the Mish function's derivative approaches zero with smaller inputs and nears one with larger inputs. This derivative behavior allows Mish to maintain robust gradient flow during training and prevent saturation with large inputs, thus facilitating effective gradient propagation and mitigating the common issue of gradient vanishing in deep networks.



(a) (b)

FIGURE 2. Comparison of two activation functions and their derivative images (a) Comparison of their functions (b) Comparison of their derivative images.

Attention mechanisms selectively highlight information relevant to a given task while reducing the influence of irrelevant data. By concentrating on the most pertinent features, these mechanisms help prevent the model from overfitting to noise and other irrelevant content in the training data.

Recently, several attention models have been developed, such as Channel Attention [15], Spatial Attention [16], and Self-Attention [17].Triplet Attention(TA) is a convolutional attention module distinguished by its tripartite structure that facilitates the capture of interactions across different dimensions to compute attention weights [18]. Notably, TA offers the following benefits: 1. Contrary to conventional attention mechanisms that focus only on channel or spatial dimensions independently, TA engages both spatial and channel dimensions simultaneously. It manipulates the input tensor by rotating it, aligning various dimensions—height, width, and channel, allowing the module to more effectively integrate information across these dimensions and generate richer, more discriminative feature representations. 2. TA is designed to be lightweight and suitable for integration in various architectures. We introduced the TA module into the network bottleneck, enhancing the information and discriminability of the feature descriptors without adding excessive parameters.

In edge computing, models encounter operational environments distinct from those experienced during training on high-performance servers. Edge devices are typically resource-constrained, imposing greater demands on the computational and storage capabilities of the models. Moreover, training and testing data may originate from diverse environments with significant differences in lighting, background, and other conditions. To adapt to these variations, models must exhibit strong robustness. To enhance the generalization robustness of models under distributional changes, CNSN normalization techniques have been incorporated into our work [19]. Cross-Norm exchanges the channel-wise mean and variance between feature maps to expand the training distribution, while Self-Norm employs attention mechanisms to recalibrate the statistics to bridging the gap between the training and testing distributions. The complementary use of Cross-Norm and Self-Norm effectively addresses the issue of poor performance in new environments, a common challenge faced by traditional normalization techniques due to distribution shifts.

The structure of the network bottleneck after the improvement is shown in Fig. 3.

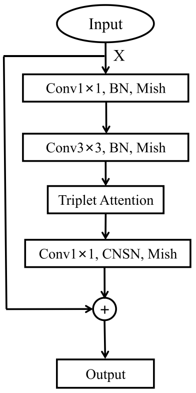


FIGURE 3. Structure of inverted residual block.

At the same time, this paper adjusts the number of layers and channels in the network to maintain model accuracy while reducing the number of parameters. Table 1 shows the network configuration of the Inverted Residual Blocks, where t is the expansion factor, n is the number of repetitions, and c denotes the number of output channels. In the TA column, a value of 1 indicates that TA is added to the inverted residual block, while a value of 0 indicates no change.

The modified network only has 5.79M parameters.

TABLE 1. Configuration of the inverted residual blocks.

Number	t	c	n	s	TA
1	2	16	2	2	0
2	6	24	3	2	1
3	6	32	3	1	0
4	6	64	4	1	0
5	6	96	2	1	1
6	6	160	3	2	0

III. MEMRISTOR-BASED NEURAL NETWORK

A. MODELS OF MEMRISTOR

In 1971, Chua introduced the concept of the memristor, defining it as the ratio of the change in magnetic flux to the change in electric charge passing through the device per unit time, expressed as:

$$v(t) = M(q(t)) \cdot i(t) \tag{1}$$

where v(t) represents the instantaneous voltage across the memristor, i(t) denotes the instantaneous current flowing through the memristor, q(t) is the total charge that has passed through the memristor up to time t, and M(q(t)) is the memristance.

1) LINEAR ION DRIFT MODEL

The current-voltage (I-V) characteristic expression of the linear ion drift model proposed by the HP research team in 2008 is

$$v(t) = \left(R_{on}\frac{w(t)}{D} + \left(R_{off}\left(1 - \frac{w(t)}{D}\right)\right)\right) i(t) \tag{2}$$

$$\frac{dw(t)}{dt} = \mu_v \times \frac{R_{on}}{D} \times i(t) \tag{3}$$

where R_{ON} and R_{OFF} represent the resistance values of the memristor in its low resistance state and high resistance state, respectively. w denotes the width of the conductive region of the memristor, D is the maximum width of the conductive region, and μ_v represents the ion mobility. The simulation of the above Linear Ion Drift Model can obtain the bipolar switching behavior and hysteresis loop as shown in Fig. 4. Where $R_{on} = 100$, $R_{off} = 16000$, $u_v = 1e^{-14}$, $D = 1e^{-08}$.

2) DATA-DRIVEN VERILOG-A RRAM

Data-Driven Verilog-A RRAM model can accurately capture the I-V characteristics of memristor devices under different types of voltage sweeps [20]. The relationship between the

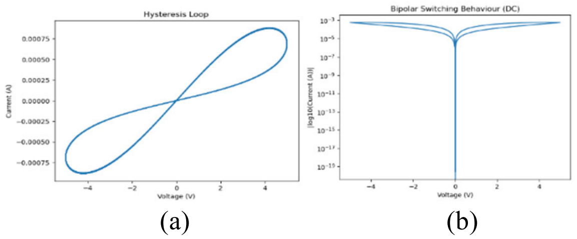


FIGURE 4. Simulation results of linear ion model (a) hysteresis loop (b) bipolar switching behavior.

current i, the bias voltage v, and the state R of the memristor can be expressed as

$$i(R,v)=\begin{cases} a_p(\frac{1}{R})sinh(b_p v), & v>0\\ a_n(\frac{1}{R})sinh(b_n v), & v\le 0\end{cases} \tag{4}$$

The rate of change of the memristor state R over time is given by the product of the switching sensitivity function s(v) and the window function f(R,v). The derivative of the memristor state R is

$$\frac{dR}{dt}=g(R,v)=s(v)\times f(R,v) \tag{5}$$

The switching sensitivity function S(v) for the change in the memristor state and window function f(R,v) are defined as

$$s(v)=\begin{cases} A_P(-1+e^{t_p|v|}) & v>0\\ 0 & v=0\\ A_n(-1+e^{t_n|v|}) & v<0\end{cases} \tag{6}$$

$$f(R,v)=\begin{cases} -1+e^{\eta k_p(r_p(v)-R)}, & R<\eta\cdot r_p(v)v>0\\ 0 & v=0\\ -1+e^{\eta k_n(r_n(v)-R)}, & R<\eta\cdot r_n(v)v<0\end{cases} \tag{7}$$

where $a_{p,n}$, $b_{p,n}$, $A_{p,n}$, $t_{p,n}$ and $k_{p,n}$ are fitting parameters. $r_{p,n}$ (v) is the voltage-dependent resistive boundary function. We set the parameters of the Data-Driven Verilog-A RRAM model as follows: $R_{on} = 1280\Omega$, $R_{off} = 1.7 \times 10^4\Omega$, $a_p = 0.24$, $a_n = 0.24$, $A_p = 743.47$, $A_n = -68012.2937$, $t_p = 6.51$, $t_n = 0.31645$, $k_p = 5.11 \times 10^{-4}$, $k_n = 1.17 \times 10^{-3}$, $r_p = [16719,0]$, $r_n = [29304.82557,23692.77225]$, $b_p = 3$, $b_n = 3$. The simulation results of RRAM model as shown in Fig. 5.

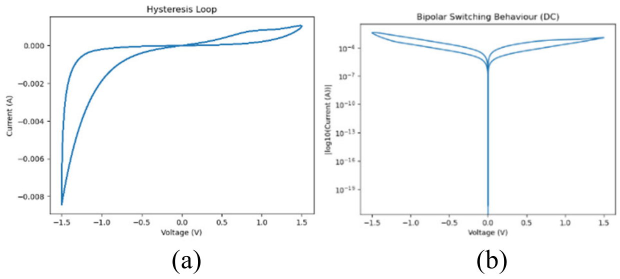


FIGURE 5. Simulation results of RRAM model (a) hysteresis loop (b) bipolar switching behavior.

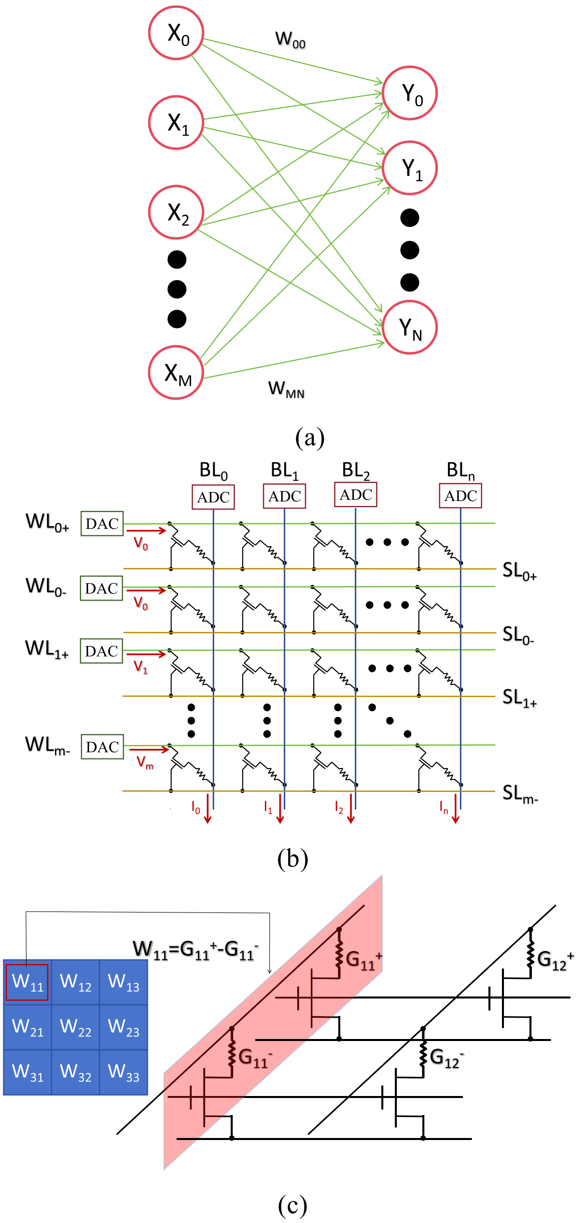


FIGURE 6. Diagram of memristor array and neural network mapping (a) Neural Networks (b) Memristor 1T1R cross array structure (c)Differential structure of memristor array.

B. MEMRISTOR CROSSBAR ARRAYS AND NEURAL NETWORKS

The memristor crossbar array can perform matrix multiplication, thereby replacing the convolutional and fully connected layers in convolutional neural networks. The schematic diagram of the mapping relationship between the memristor crossbar array and the neural network is shown in Fig. 6. Fig. 6(a) represents the structure of a single-layer feedforward neural network, where X_0 to X_M denote the inputs, Y_0 to Y_N denote the outputs, and W_{00} to W_{MN} denote the weights between the two layers of neurons. The computational relationship between the two feedforward network layers is expressed as

$$[Y_0 \cdots Y_N] = [X_0 \cdots X_M] \begin{bmatrix} W_{00} \cdots W_{0N} \\ \cdots\cdots\cdot \\ W_{M0} \cdots W_{MN} \end{bmatrix} \tag{8}$$

The input and weight matrices undergo vector multiplication to ultimately produce the output, which is a commonly used operation in neural networks. The memristor array in Fig. 6(b) adopts a single-transistor-single-memristor (1T1R) structure. The input voltages V_0 to V_M on the word lines (WL) correspond to the inputs X_0 to X_M in the feedforward neural network shown in Fig. 6(a). After passing through the transistors and memristors, the output currents I_0 to I_N on the bit lines (BL) correspond to the outputs Y_0 to Y_N in the feedforward neural network. The conductance values g_{00} to g_{MN} of the memristor crossbar array correspond to the weights W_{00} to W_{MN} between the two layers of neurons. The DAC generates precise analog voltages to control the memristor states, while the ADC converts the memristor's analog resistance values into digital signals. The input-output relationship of the memristor array can be denoted by

$$[I_0 \cdots I_N] = [V_0 \cdots V_M] \begin{bmatrix} g_{00} \cdots g_{0N} \\ \cdots\cdots \\ g_{M0} \cdots g_{MN} \end{bmatrix} \tag{9}$$

Since a single conductance can't represent negative weights, the weight is represented by the difference in conductance between two adjacent 1T1R cells, as shown in Fig. 6(c).

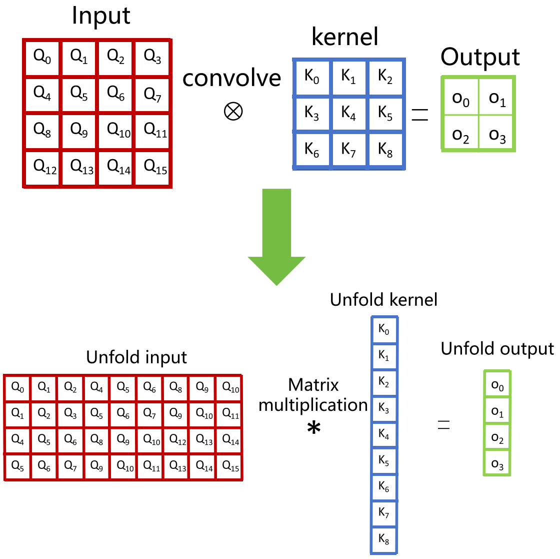


FIGURE 7. The process of mapping convolutional layers onto memristor arrays.

The convolution process of the memristor crossbar array is illustrated in Fig. 7. Assume the input is a 4 × 4 image and the convolution kernel size is 3 × 3. After performing convolution on the input image, the output size is 2 × 2. The input image is unfolded and then matrix-multiplied with the unfolded convolution kernel to obtain the output results. Here, $[Q_0,Q_1,\ldots,Q_{15}]$ correspond to the input voltages $[V_0,V_1,..,V_{15}]$ on the word lines of the memristor array, the convolution kernel $[K_1,K_2,\ldots,K_8]$ corresponds to the conductance values g of the memristors, and $[O_1,O_2,O_3,O_4]$ correspond to the output currents $[I_1, I_2, I_3, I_4]$ on the bit line.

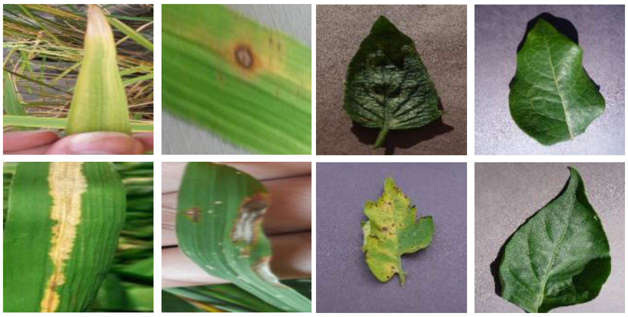


FIGURE 8. Examples of the datasets (a) PV (b) RLD.

TABLE 2. Ablation experiment.

Net	Params Size/ MB	Accuracy Rate /%	
		PV	RLD
MobileNetV2	8.74	97.33	97.81
MobileNetV2(modified)	5.73	97.02	96.88
MobileNetV2(modified)+TA	5.74	98.86	98.82
MobileNetV2(modified)+CNSN	5.78	97.65	97.57
TTN-MobileNetV2(ours)	5.79	99.03	99.16

TABLE 3. Comparative experiment.

Net	Params Size/	Accuracy Rate/%	
	MB	PV	RLD
MobileNetV2	8.74	97.33	97.81
MobileNetV3-large[23]	16.14	96.55	96.43
ResNet50[24]	89.91	97.65	98.06
VGG16[25]	512.23	91.97	97.23
TTN-MobileNetV2(ours)	5.79	99.03	99.16

IV. EXPERIMENTAL METHOD AND RESULTS ANALYSIS

A. DATASET SELECTION

The publicly available Plant Village(PV) and Rice Leaf Disease(RLD) datasets are used as the training and testing datasets. PV [21] is a dataset widely used for evaluating plant disease recognition and research models. The PV dataset contains 54,305 high-quality leaf images, covering 14 different crops, 38 different disease categories which have high resolution and mostly white backgrounds, facilitating disease identification. The RLD dataset is an image dataset specifically used for the detection and classification of rice leaf diseases. It includes 5,932 images of three common rice diseases and a healthy category: Brown Spot, Leaf Blast, Bacterial Blight, and Healthy. The widespread use of this dataset can improve the automation and intelligence of rice disease management. In our study, 80% of the data is used for the training set, and the remaining 20% is used for testing.

Fig. 8 represents some samples from the PV and RLD datasets.

B. EXPERIMENTAL PROCESS

The research is conducted in an operating environment of Ubuntu 22.04, with Python version 3.11.4, PyTorch version

TABLE 4. Testing results of memristor-based network using different activation functions.

Activation function	Original network accuracy rate/%		Linear Ion Drift accuracy rate/%		Data-driven accuracy rate/%	
	PV	RLD	PV	RLD	PV	RLD
SiLU	98.67	98.99	96.91	93.26	97.75	93.42
ReLU6	98.38	98.65	89.06	91.15	92.29	92.75
Mish	99.03	99.16	98.71	98.64	98.69	98.55
GeLU	98.42	98.82	98.12	96.71	98.21	96.29

2.1.0, and a NVIDA GeForce GTX4090D GPU. Firstly, the improved model TTN-MobileNetV2 is trained using the PV and RLD datasets. The experiment involves 60 training epochs, and the network parameters are optimized using the SGD algorithm with an initial learning rate of 0.001. Every 20 epochs, the learning rate decays to one-tenth of its original value to achieve finer optimization steps at different training stages. Cross-entropy is used as the loss function for the network. For evaluating the classification model, accuracy and parameter count serve as the evaluation metrics.

The improved MobileNetV2 model's convolutional and fully connected layers are re-encoded using the application interface (API) provided by the MemTorch [22], mapping all network model weights to the conductance values of memristors in the memristor crossbar array. Input signals are scaled to −0.3 to +0.3 to simulate voltage signals from −0.3V to +0.3V. The weights of all convolutional and fully connected layers in the saved best weight file are mapped to a 128*128 memristor crossbar array, thereby constructing a memristor convolutional neural network. The constructed network is evaluated using accuracy, and the test results are compared with the original network.

C. TEST EXPERIMENT OF TTN-MobileNetV2

This paper modifies the configuration of layers and channels in the traditional model to reduce the number of parameters, while employing attention mechanisms, CNSN normalization layers, and Mish activation functions to enhance the model's feature extraction capabilities. We conducted comparative ablation studies on the PV and RLD datasets to analyze the impact of different modifications on the network model. The results of the ablation experiments are shown in Table 2, where MobileNetV2(modified) represents the model with changes only to the channel number and layer depth of the inverted residual blocks. In both leaf disease recognition datasets, the network with modified layer depth and channel number showed a significant reduction in parameter count, but accuracy did not decrease dramatically. Attention mechanisms and CNSN both improve the model's performance without significantly increasing the parameter size. Finally, the improved model outperformed the baseline model by 1.7% and 1.35% on the two datasets, respectively.

We further validated our method by comparing it with mainstream convolutional neural networks. The comparison results are shown in Table 3. From experimental results, it indicates that the proposed network is advanced in terms of both the number of parameters and accuracy.

D. TEST EXPERIMENT OF MEMRISTOR-BASED NEURAL NETWORK

After the network training is completed, we use the Memtorch API to map the convolutional layers and fully connected layers of the convolutional neural network to the corresponding memristor array. Specifically, the process includes the following steps: (1) Select and initialize the memristor model. In this paper, data-driven Verilog-A RRAM and linear ion drift model are selected as the basis for simulation calculations. (2) Load the TTN- MobileNetV2 model as the test model. (3) Define a mapping configuration that maps the weights of convolutional neural networks to the conductivity values of memristors. (4) Convert the weights of convolutional layers and fully connected layers into corresponding conductivity values, and assign these conductivity values to the memristor array to simulate convolutional operations and matrix multiplication. (5) Evaluate the performance of the model by using a test dataset. Follow the above steps to map TTN-MobileNetV2 to a memristor array and test it on PV and RLD datasets. The experimental results are shown in Table 4. It can be seen that the activation function in the network model will affect the simulation characteristics of the memristor array, thereby affecting the final test results. In addition, when the range and linearity of the conductance value of the memristor do not match the output range and distribution characteristics of certain activation functions, nonlinear errors and noise will be introduced, which can also affect the accuracy and stability of the simulation results. The results showed that both the original network model and the convolutional neural network model based on memristors achieved the highest testing accuracy when using Mish as the activation function. On the PV and RLD datasets, the accuracy losses of the two memristor models are 0.32, 0.34, and 0.52, 0.61, respectively.

V. CONCLUSION

The MobileNetV2 network architecture is analyzed in this paper, then the Triplet attention mechanism is introduced and the CNSN normalization techniques are used to improve the MobileNetV2 network, enhancing the generalization robustness and recognition accuracy of the MobileNetV2 network. The accuracy of this network for leaf disease recognition tasks on PV and RLD datasets is 99.03% and 99.16%, respectively. Based on the Mem-Torch framework, two types of memristor models, Data Driven Verilog-A RRAM and linear ion drift model, were used to test the TTN-MobileNetV2 memristor network. The recognition accuracy losses of the corresponding TTN-MobileNetV2 memristor networks for the two models were 0.32, 0.34, and 0.52, 0.61, respectively, which can meet the performance requirements in the field of plant leaf disease recognition. Memristor convolutional neural networks, with their inherent advantages of high speed and low power consumption, will undoubtedly be widely applied in the fields of plant disease management and agricultural ecological balance protection.

REFERENCES

[1] M. Shoaib, B. Shah, S. EI-Sappagh, A. Ali, A. Ullah, F. Alenezi, T. Gechev, T. Hussain, and F. Ali, “An advanced deep learning models-based plant disease detection: A review of recent research,” *Frontiers Plant Sci.*, vol. 14, Mar. 2023, Art. no. 1158933.

[2] J. Andrew, J. Eunice, D. E. Popescu, M. K. Chowdary, and J. Hemanth, “Deep learning-based leaf disease detection in crops using images for agricultural applications,” *Agronomy*, vol. 12, no. 10, p. 2395, Oct. 2022.

[3] S. H. Lee, H. Goëau, P. Bonnet, and A. Joly, “New perspectives on plant disease characterization based on deep learning,” *Comput. Electron. Agricult.*, vol. 170, Mar. 2020, Art. no. 105220.

[4] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, “Deep feature based rice leaf disease identification using support vector machine,” *Comput. Electron. Agricult.*, vol. 175, Aug. 2020, Art. no. 105527.

[5] M.-J. Lee, C. B. Lee, D. Lee, S. R. Lee, M. Chang, J. H. Hur, Y.-B. Kim, C.-J. Kim, D. H. Seo, S. Seo, U.-I. Chung, I.-K. Yoo, and K. Kim, “A fast, high-endurance and scalable non-volatile memory device made from asymmetric Ta_2O_{5-x}/TaO_{2-x} bilayer structures,” *Nature Mater.*, vol. 10, no. 8, pp. 625–630, Aug. 2011.

[6] J. Zhou, F. Cai, Q. Wang, B. Chen, S. Gaba, and W. D. Lu, “Very low-programming-current RRAM with self-rectifying characteristics,” *IEEE Electron Device Lett.*, vol. 37, no. 4, pp. 404–407, Apr. 2016.

[7] A. C. Torrezan, J. P. Strachan, G. Medeiros-Ribeiro, and R. S. Williams, “Sub-nanosecond switching of a tantalum oxide memristor,” *Nanotechnology*, vol. 22, no. 48, Dec. 2011, Art. no. 485203.

[8] W. Chen, Z. Qi, Z. Akhtar, and K. Siddique, “Resistive-RAM-based in-memory computing for neural network: A review,” *Electronics*, vol. 11, no. 22, p. 3667, Nov. 2022.

[9] W. Xu, J. Wang, and X. Yan, “Advances in memristor-based neural networks,” *Frontiers Nanotechnol.*, vol. 3, Mar. 2021, Art. no. 645995.

[10] M. A. Zidan, J. P. Strachan, and W. D. Lu, “The future of electronics based on memristive systems,” *Nature Electron.*, vol. 1, no. 1, pp. 22–29, Jan. 2018.

[11] M. Le Gallo, A. Sebastian, R. Mathis, M. Manica, H. Giefers, T. Tuma, C. Bekas, A. Curioni, and E. Eleftheriou, “Mixed-precision in-memory computing,” *Nature Electron.*, vol. 1, no. 4, pp. 246–253, Apr. 2018.

[12] P. Yao, H. Wu, B. Gao, J. Tang, Q. Zhang, W. Zhang, J. J. Yang, and H. Qian, “Fully hardware-implemented memristor convolutional neural network,” *Nature*, vol. 577, no. 7792, pp. 641–646, Jan. 2020.

[13] W. Zhang, P. Yao, B. Gao, Q. Liu, D. Wu, Q. Zhang, Y. Li, Q. Qin, J. Li, Z. Zhu, Y. Cai, D. Wu, J. Tang, H. Qian, Y. Wang, and H. Wu, “Edge learning using a fully integrated neuro-inspired memristor chip,” *Science*, vol. 381, no. 6663, pp. 1205–1211, Sep. 2023.

[14] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[15] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[16] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional block attention module,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 6000–6010.

[18] D. Misra, T. Nalamada, A. U. Arasanipalai, and Q. Hou, “Rotate to attend: Convolutional triplet attention module,” in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3138–3147.

[19] Z. Tang, Y. Gao, Y. Zhu, Z. Zhang, M. Li, and D. Metaxas, “CrossNorm and SelfNorm for generalization under distribution shifts,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 52–61.

[20] I. Messaris, A. Serb, S. Stathopoulos, A. Khiat, S. Nikolaidis, and T. Prodromakis, “A data-driven Verilog-A ReRAM model,” *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 37, no. 12, pp. 3151–3162, Dec. 2018.

[21] D. P. Hughes and M. Salathe, “An open access repository of images on plant health to enable the development of mobile disease diagnostics,” 2015, *arXiv:1511.08060*.

[22] C. Lammie, W. Xiang, B. Linares-Barranco, and M. R. Azghadi, “MemTorch: An open-source simulation framework for memristive deep learning systems,” *Neurocomputing*, vol. 485, pp. 124–133, May 2022.

[23] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, “Searching for MobileNetV3,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324.

[24] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[25] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, *arXiv:1409.1556*.

NENGYUAN PAN received the B.Tech. degree in electronic information science and technology from Hubei University of Education, Wuhan, China, in 2023. He is currently pursuing the master’s degree with the School of Artificial Intelligence, Hubei University, Wuhan.

His research interests include deep learning, machine learning, and memristor neural network image processing.

WEIMING YANG received the B.Tech. degree in radio technology from Beijing Institute of Technology, Beijing, China, in 1990, the M.S. degree in communication and information systems from the Huazhong University of Science and Technology, Wuhan, China, in 2001, and the Ph.D. degree in microelectronics and solid-state electronics from Beijing University of Technology, Beijing, in 2006.

His research interests include RF devices and circuit integration, microwave sensing, and information processing.

YUTING LUO received the B.Tech. degree in rail transportation signaling and control from Hubei University of Science and Technology, Wuhan, China, in 2020. She is currently pursuing the master’s degree with the School of Artificial Intelligence, Hubei University, Wuhan.

Her research interests include deep learning, machine learning, and neural networks.

YONGLIN WANG received the B.Tech. degree in electronic information science and technology from Hubei University of Education, Wuhan, China, in 2023. He is currently pursuing the master’s degree with the School of Artificial Intelligence, Hubei University, Wuhan.

His research interests include deep learning, machine learning, neural networks, and grayscale image coloring.