

Research Article

An Electronic Component Recognition Algorithm Based on Deep Learning with a Faster SqueezeNet

Yuanyuan Xu ^{1,2}, Genke Yang ^{1,3}, Jiliang Luo,² and Jianan He⁴

¹Department of Automation, Shanghai Jiaotong University, Shanghai 200240, China

²College of Information Science and Engineering, Huaqiao University, Xiamen 361021, China

³NingBo Artificial Intelligence Institute, Shanghai Jiaotong University, Ningbo 315000, China

⁴Central Laboratory of Health Quarantine, Shenzhen International Travel Health Care Center, Shenzhen Academy of Inspection and Quarantine, Shenzhen Customs District, Shenzhen 518033, China

Correspondence should be addressed to Genke Yang; gkyang@sjtu.edu.cn

Received 9 July 2020; Revised 20 August 2020; Accepted 28 September 2020; Published 28 October 2020

Academic Editor: Chenxi Huang

Copyright © 2020 Yuanyuan Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Electronic component recognition plays an important role in industrial production, electronic manufacturing, and testing. In order to address the problem of the low recognition recall and accuracy of traditional image recognition technologies (such as principal component analysis (PCA) and support vector machine (SVM)), this paper selects multiple deep learning networks for testing and optimizes the SqueezeNet network. The paper then presents an electronic component recognition algorithm based on the Faster SqueezeNet network. This structure can reduce the size of network parameters and computational complexity without deteriorating the performance of the network. The results show that the proposed algorithm performs well, where the Receiver Operating Characteristic Curve (ROC) and Area Under the Curve (AUC), capacitor and inductor, reach 1.0. When the FPR is less than or equal 10^{-6} level, the TPR is greater than or equal to 0.99; its reasoning time is about 2.67 ms, achieving the industrial application level in terms of time consumption and performance.

1. Introduction

The scientific and standardized classification of electronic components is important for enterprise informatization and improving the management of electronic components. It is not only essential for designers, management departments, and purchasing departments, where they need to investigate and use products accurately and efficiently, but also helpful for the management of engineering components. Many institutions attach great importance to the procurement of components, based on their own needs, and the establishment of a component classification system.

As an example, NASA's NPSL (NASA Parts Selection List) [1] divides electronic components into 10 major categories and 28 minor categories based on their function. The U.S. military classifies the QPL list into 264 subcategories based on the specific functions of the electronic components. Electronic components published by the U.S. Defense

Supply Center (DSCC) are classified into 37 categories according to their military drawing component number [2]. The European Space Agency's (ESA) PPL list covers 15 major categories and 67 minor categories of electronic components, while the QPL list covers 14 major categories and 37 minor categories of electronic components. The classification table of GB/T 17564.4-2001 is given where components are classified into three parts: Electric-Electronic (EE), Electro-Mechanical (EM), and magnetic parts, including amplifiers, antennas, batteries, capacitors, conductors, and another 31 categories [3–5].

Electronic components play an important role in the development of industry. There are many kinds of electronic components, and these are developing towards miniaturization and chips. In the production, scientific research, application, and recycling of electronic components, classification is a very important fundamental task. Therefore, it is of great practical significance to construct an automatic

identification system for electronic components which can operate in real time [6].

Because traditional image recognition technologies (such as principal component analysis (PCA) and support vector machine (SVM)) have the problem of low recall and accuracy, this paper selects lightweight deep learning networks for testing and draws on the structure of DenseNet and ResNet to optimize the SqueezeNet network. An electronic component identification system based on the Faster SqueezeNet algorithm is proposed. This structure can reduce the size of network parameters and computational complexity without reducing network performance. Experimental results show that the proposed Faster SqueezeNet algorithm has excellent performance in terms of model parameters, inference time, and portability and is suitable for engineering applications of electronic component recognition in electronic manufacturing [7–9].

2. Related Work

Current image classification methods are mostly divided into two categories [10]. The first category mainly classifies images based on image spatial domains and transform domains. The second category classifies images using deep learning Networks such as convolutional neural networks (CNNs) to automatically learn image features.

2.1. Traditional Image Classification Methods. Traditional image classification methods have been studied and developed for many years. These methods first extract image features through a series of complex image preprocessing steps (such as morphological transforms [11]) and then classify them according to those features [12–14]. For example, Du et al. [15] combined the least squares method with the Hough transform and classified components by extracting the edge features of components [16–18]. Although the classification accuracy is relatively high, traditional image classification methods cannot process very large images [19], the computational complexity is prohibitive, and it is difficult to achieve multiple component classification at the same time.

2.2. Deep Learning Network-Based Recognition Methods. Convolutional neural networks, [20–23] integrated with convolutions, backpropagation, weight sharing, sparse connections, pooling, and other ideas, can automatically extract image features. This addresses the problem of processing very large images and reaches the level of industrial application concerning the problem of multiple classification [24–26]. However, general deep learning networks have a complex structure and a large number of calculations, which is incompatible with large amounts of data and reasonable real-time performance.

In this paper, we study deep learning network models such as ResNet, SqueezeNet, DenseNet, MobileNetV2, and EfficientNet [27–30]. Based on the analysis of traditional methods and deep learning methods, the improved Faster SqueezeNet is chosen as the network model for electronic

component classification. After further modification of its network structure, the Faster SqueezeNet can meet the real-time and accuracy requirements of industrial electronic component identification. The TPR is sufficiently high and an acceptable FPR is guaranteed, that is, the AUC is large enough. At the same time, the number of parameters of the Faster SqueezeNet is further reduced and the processing time is improved. Figure 1 shows a diagram of electronic component classification methods.

3. The Dataset

All the training, verification, and test images in this paper are based on printed circuit boards of cell phones, computers, air conditioners, automobiles, etc. All the data are obtained by template matching, PS matting, circuit board CAD information, and other approaches. Solder paste detection (SPI) equipment is used, which includes more than 30 thousand pictures with 8000×9000 pixels. A partial image of the whole board is shown in Figures 2 and 3.

The environment of the circuit board is single and closed, so there is no complex image preprocessing in this paper [31–36]. The main preprocessing includes left and right mirror images, upper and lower mirror images, random noise, random brightness, random rotation of 30, 90, 120, and 180 degrees, and boundary filling, as shown in Figure 4.

4. The Architecture

A CNN is a feedforward neural network, which can extract features from a two-dimensional image and optimize parameters in the network using backpropagation. Many classification network models are considered in this paper, of which SqueezeNet and Faster SqueezeNet are superior.

4.1. SqueezeNet. Because the number of parameters for AlexNet and VGGNet are getting larger and larger, the SqueezeNet network model was proposed that has very few parameters while preserving the accuracy [37, 38]. The Fire module is the core base module in SqueezeNet, and its structure is shown in Figure 5. This module is divided into Squeeze and Expand structures. Squeeze contains $S \ 1 \times 1$ convolution kernels. The Expand layer contains 1×1 kernels and 3×3 convolution kernels. The number of 1×1 convolution kernels is $E_{1 \times 1}$, and the number of 3×3 convolution kernels is $E_{3 \times 3}$. The model must satisfy $S < (E_{1 \times 1} + E_{3 \times 3})$.

The 1×1 convolutional layer has attracted much attention in the consideration of network structures. Min used a multilayer perceptron (MLP) instead of the traditional linear convolution kernel to improve the expressive power of the network [39]. The work also explains from the perspective of cross-channel pooling that MLP is equivalent to the cascaded cross-channel parametric pooling layer behind the traditional convolution kernel, thus achieving a linear combination of multiple feature maps and information integration across channels.

When the number of input and output channels is large, the convolution kernel parameter becomes large. We add a 1×1 convolution to every inception module, reducing the

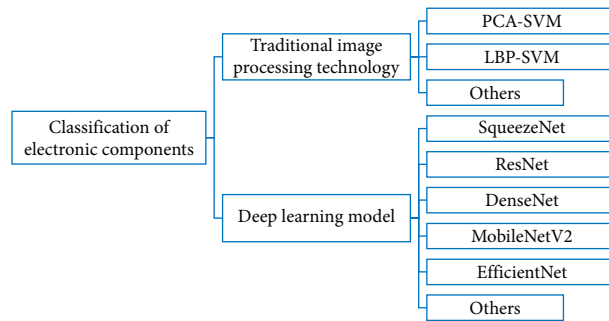


FIGURE 1: Electronic component classification methods.

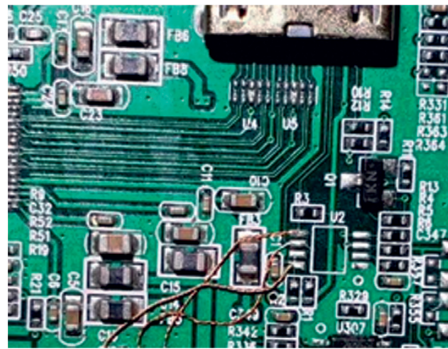


FIGURE 2: Partial image of the whole PCB.

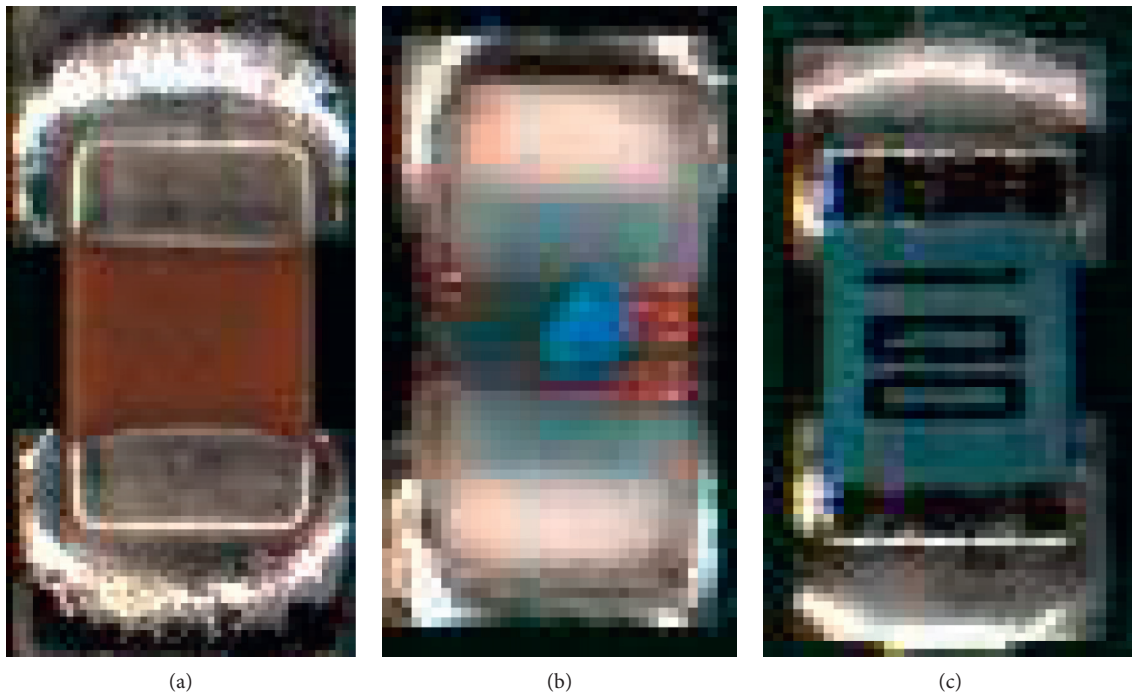


FIGURE 3: Images of the electronic components. (a) The capacitor. (b) The inductor. (c) The resistor.

number of input channels, and the convolution kernel parameters and operation complexity also decrease [40]. At the end of the structure, a 1×1 convolution is added to improve the number of channels and enhance feature extraction [41].

SqueezeNet replaces 3×3 convolutions with 1×1 convolutions to reduce the number of parameters to one-ninth. When the sampling reduction operation is delayed, a larger activation graph can be provided for the convolutional layer,

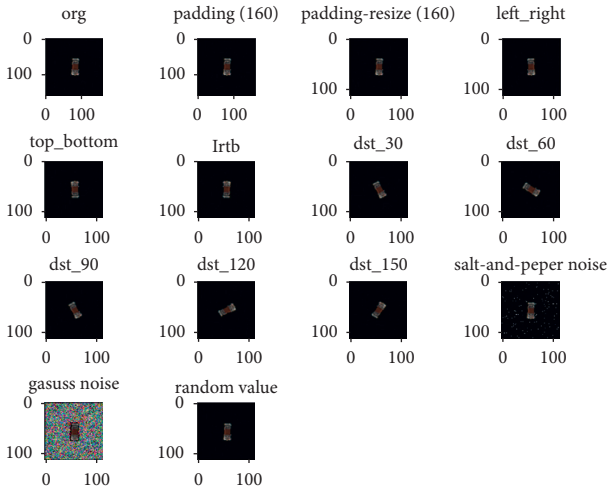


FIGURE 4: Results of different image preprocessing steps.

while the larger activation graph retains more information, which can provide a higher classification accuracy.

4.2. Faster SqueezeNet. In order to improve the accuracy and real-time performance of electronic component classification, Faster SqueezeNet is proposed. In order to prevent overfitting, we added BatchNorm and residual structures. At the same time, like DenseNet, we use concat to connect different layers to enhance the expressiveness of the first few layers in the network.

Faster SqueezeNet consists of 1 BatchNorm layer, 3 block layers, 4 convolution layers, and a global average pooling layer. The model is shown in Figure 6.

Faster SqueezeNet is mainly improved in the following ways:

- (1) In order to further improve the information flow between layers, we imitated the DenseNet structure and proposed a different connection mode [42]. This consists of a pooling layer and a fire module, and finally the two concat layers are connected to the next convolution layer. Its structure is shown in the green dotted box in Figure 6.

The current layer receives all the feature maps of the previous layer, and we use x_0, \dots, x_{l-1} as input; then, x_l is shown as

$$x_l = H_l([x_0, x_1, \dots, x_{l-1}]), \quad (1)$$

where $[x_0, x_1, \dots, x_{l-1}]$ refers to the connection of feature graphs generated in layer $0, 1, \dots, l-1$ and $H_l(\cdot)$ concatenates multiple inputs. Here, x_0 represents the max pooling layer, x_1 represents the Fire layer, and x_l is the concat layer.

Without excessively increasing the number of network parameters, the performance of the network is enhanced in the early stages, and at the same time,

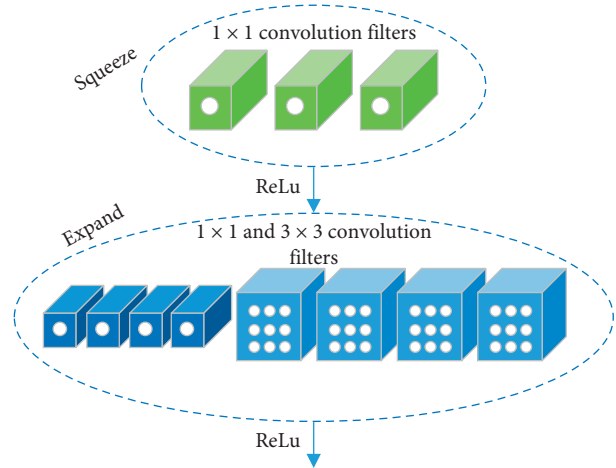


FIGURE 5: The Fire module.

any two-layer network can directly communicate information.

- (2) In order to ensure better network convergence, we learn from the ResNet structure and propose different building blocks which consist of a pooling layer and a fire module. Finally, after the two layers are summed, it is connected to the next convolutional layer. Its structure is shown in the green dotted box in Figure 6.

In ResNet, the shortcut connection directly uses identity mapping, which means that the input of a convolution stack is directly added to the output of the convolution stack [39]. Formally, denoting the desired underlying mapping as $H(x)$, we let the stacked nonlinear layers fit another mapping of $F(x) := H(x) - x$. The original mapping is recast into $F(x) + x$. $F(x) + x$ can be realized by a structure called a shortcut connection in the actual encoding process. The shortcut connection usually skips one or more layers. Therefore, we use the residual structure of ResNet for reference to address the problem of vanishing gradients and degradation without increasing the number of network parameters. The residual structure is shown in Figure 7.

The model parameters for Faster SqueezeNet are shown in Table 1.

5. Experiments

All experimentation in this paper is conducted using the Windows platform. The machine uses an Intel Core i5-7300H 2.5 GHz CPU, an NVIDIA GTX1050 (2G) GPU, and 8 Gb memory. In addition to selecting several deep learning networks such as ResNet [23], SqueezeNet, MobleNet V2 [24, 27], DenseNet [28], and EfficientNet [29], this paper also uses the traditional PCA + SVM algorithm for comparison.

This paper mainly classifies resistor, capacitor, and inductor in electronic components. Specifically, 0–6 and 11–17 are 14 types of capacitors; 7–9 and 18–20 are six kinds of resistors; 10 and 21 are two types of inductors; and a total of 22 small classes. In the experimentation, the training and validation set totaled 40,000 pictures; 50% were used for

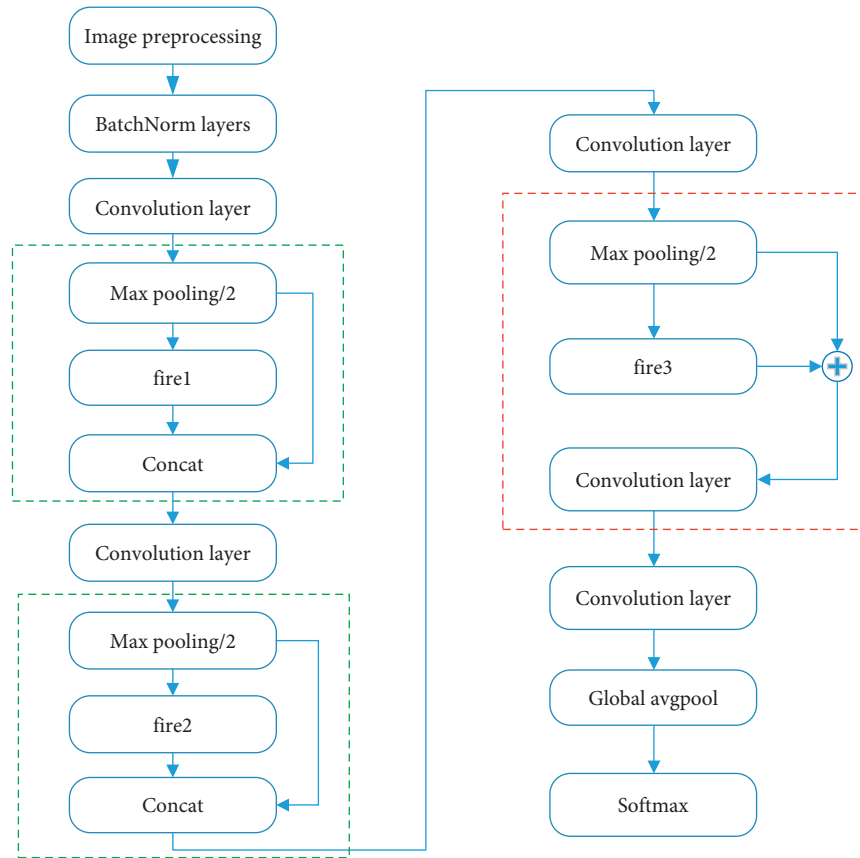


FIGURE 6: The Faster SqueezeNet network structure.

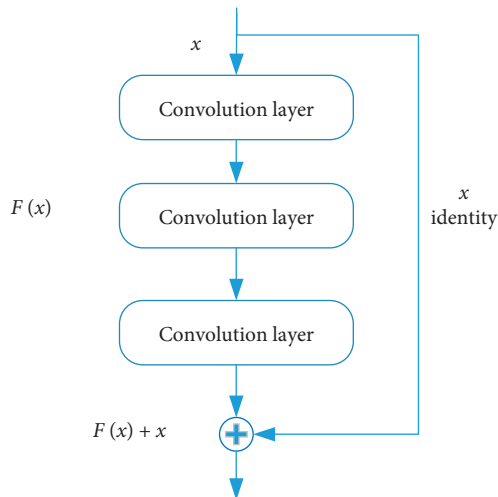


FIGURE 7: The residual structure.

training and 50% for validation. Each category (resistor, capacitor, and inductor) has 100–1000 pictures, and the test set has 10,264 pictures.

5.1. PCA + SVM. When the principal component dimensions are 32, 128, 256, and 512, the average value of each index increases with the increase of components. When the

principal component dimension is 128, the ROC curve is the best, as shown in Figure 8.

For the ROC curve, the false positive rate (FPR) is the horizontal axis, which represents the rate of false positives among all samples that are actually negative. The true positive rate (TPR) is the vertical axis, which represents the ratio of all positive samples that are correctly judged as positive. The different colored lines represent the ROC performance for different electronic components.

As can be seen from Figure 8, when $FPR \leq 0.01$, the TPR declines rapidly. In application, when the number of components increases to a certain scale, the performance of the model will be seriously degraded, and various misjudgments will often occur.

5.2. Deep Learning Algorithm Results. When the backbone network uses EfficientNet, MobleNet V2, SqueezeNet, and Faster SqueezeNet (our approach), the obtained ROC curves are shown in Figures 9–12.

The parameters and runtimes of these models are shown in Table 2.

As can be seen from Table 2, when the FPR is $10e-6$, the performance of PCA + SVM decreases linearly, while deep learning methods still have good performance. By further improving Faster SqueezeNet, our model size is substantially reduced.

TABLE 1: Model parameters for faster SqueezeNet.

Layer name/type	Output size	Filter size/stride	Depth	$S_{1 \times 1}$ (squeeze)	$E_{1 \times 1}$ (expand)	$E_{3 \times 3}$ (expand)
Input	112 * 112 * 3	—	—	—	—	—
BatchNorm	112 * 112 * 3	—	0	—	—	—
Conv 1	112 * 112 * 8	3 * 3/1	1	—	—	—
Max pool 1	56 * 56 * 1	2 * 2/2	0	—	—	—
Fire 1	56 * 56 * 32	—	2	8	16	32
Concat 1	56 * 56 * 48	—	0	—	—	—
Conv 2	56 * 56 * 48	3 * 3/1	1	—	—	—
Max pool 2	28 * 28 * 16	2 * 2/2	0	—	—	—
Fire 2	28 * 28 * 32	—	2	8	16	32
Concat 2	28 * 28 * 48	—	0	—	—	—
Conv 3	28 * 28 * 48	3 * 3/1	1	—	—	—
Max pool 3	14 * 14 * 16	2 * 2/2	0	—	—	—
Fire 3	14 * 14 * 32	—	2	8	16	32
Add	14 * 14 * 48	—	0	—	—	—
Conv 4	14 * 14 * 48	3 * 3/1	1	—	—	—
Conv 5	7 * 7 * 48	3 * 3/2	1	—	—	—
Global avgpool	1 * 1 * 22	7 * 7/7	0	—	—	—
Softmax	22	—	—	—	—	—

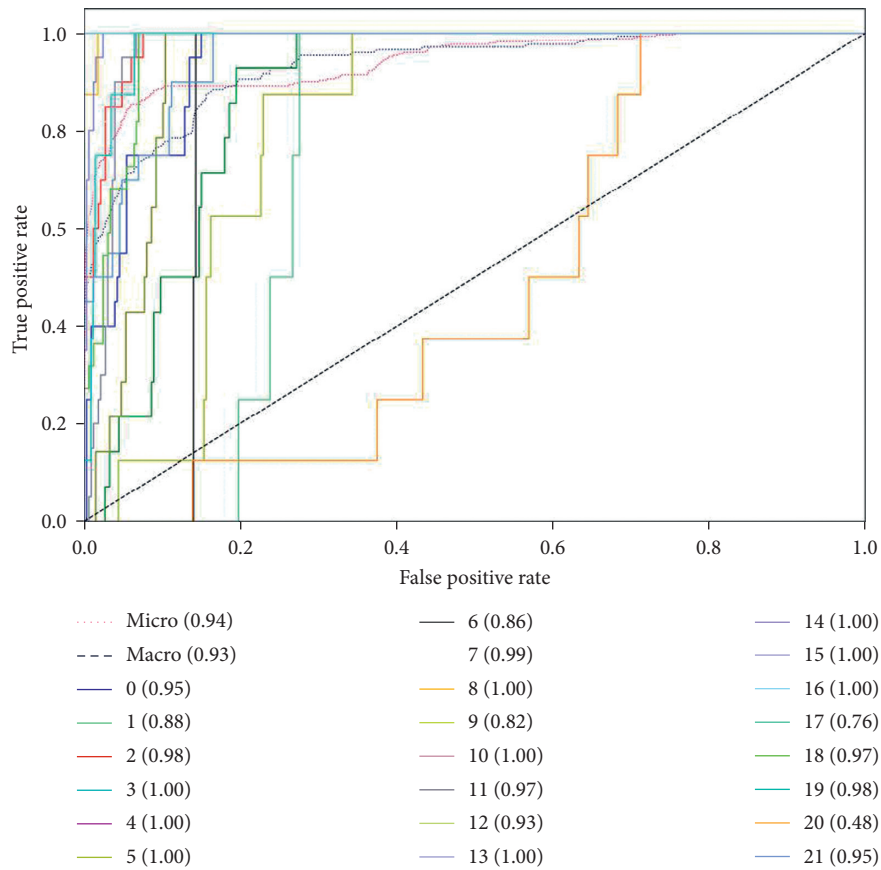


FIGURE 8: PCA and SVM ROC curve.

At the same time, TensorRT-based applications perform up to 40 times faster than CPU-only platforms during inference [43]. TensorRT can optimize all neural network models trained in the main framework to calibrate low accuracy with high accuracy [44]. It can also be

deployed to large-scale data centers, embedded or automotive products, and other platforms. Therefore, model inference is accelerated again by TensorRT to meet the requirements of real-time electronic component identification.

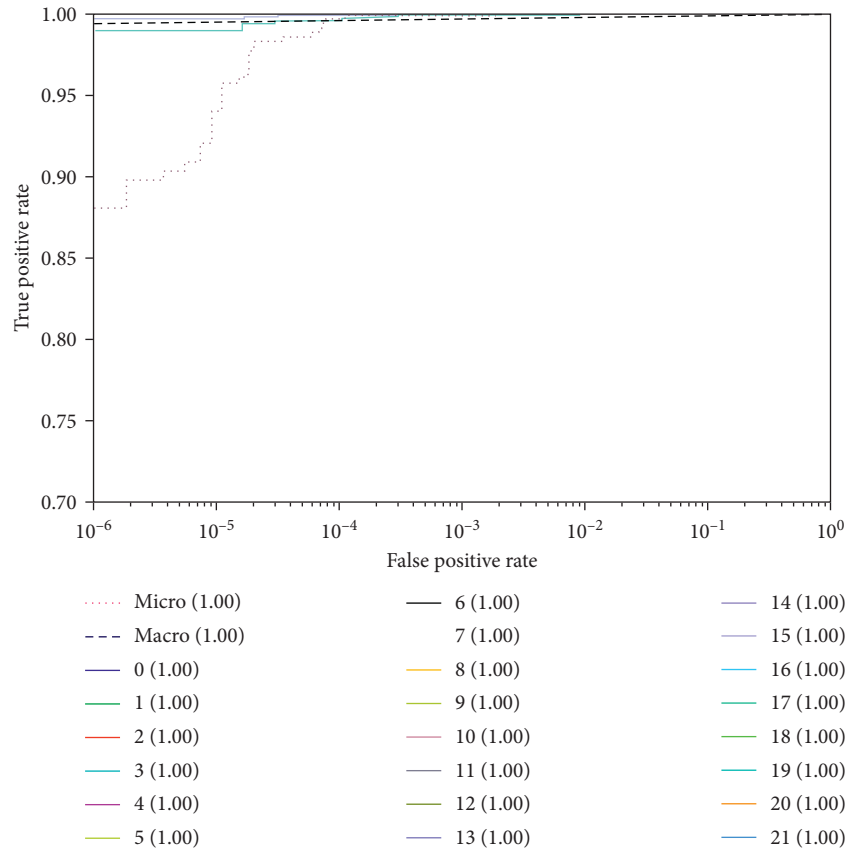


FIGURE 9: EfficientNet ROC curve.

6. Evaluation

Through the above experiments, we can summarize the performance of the traditional algorithm and deep learning algorithms.

6.1. PCA + SVM. Du et al. [15] used the least squares algorithm and the Hough transform to extract the edge of components. The classification accuracy in the three categories of resistor, capacitor, and inductor was 87%, 82.6%, and 100%, respectively. Although the result is good, this still lags behind industrial application. PCA combined with SVM is a very good and commonly used traditional method for the classification of electronic components. When the dimension of the eigenvectors increased from 22 to 880, the precision of some categories decreased. Of course, when the number of eigenvector dimensions is equal to the number of images in the training set, the time consumption also doubles (compared with 1024 eigenvector dimensions). Therefore, it is difficult to reach the industrial application level.

6.2. Deep Learning Networks. For each category, the ROC curve area for the Faster SqueezeNet algorithm can always approach 1.0, and the TPR can achieve above 0.999996 when the FPR is below 10^{-6} . Although its time consumption

(2.57 ms to 4.85 ms) is relatively high compared with the traditional method, the average time of 1000 reasoning times is about 0.68 ms after TensorRT acceleration, which can meet the requirements of practical applications. The graphics card is a GTX 1050 (2 Gb memory), which can be used on most computers.

The experimental results show that the proposed Faster SqueezeNet algorithm can identify the electronic components on the circuit board, and the improved model has good robustness and application potential. Compared with the traditional method or other deep learning models, the Fast SqueezeNet proposed in this paper has excellent performance concerning the parameters, reasoning time, and portability of the model, which is suitable for engineering applications of electronic component identification in electronic manufacturing.

7. Discussion

In this paper, the electronic components are classified into 22 subcategories of resistor, capacitor, and inductor via an improved SqueezeNet. The improved SqueezeNet draws on the structure of DenseNet and ResNet, the number of parameters is reduced, and the speed becomes faster.

However, there are thousands of other different types of resistors, capacitors, and inductors, such as integrated circuits. Moreover, there are other types of components. When

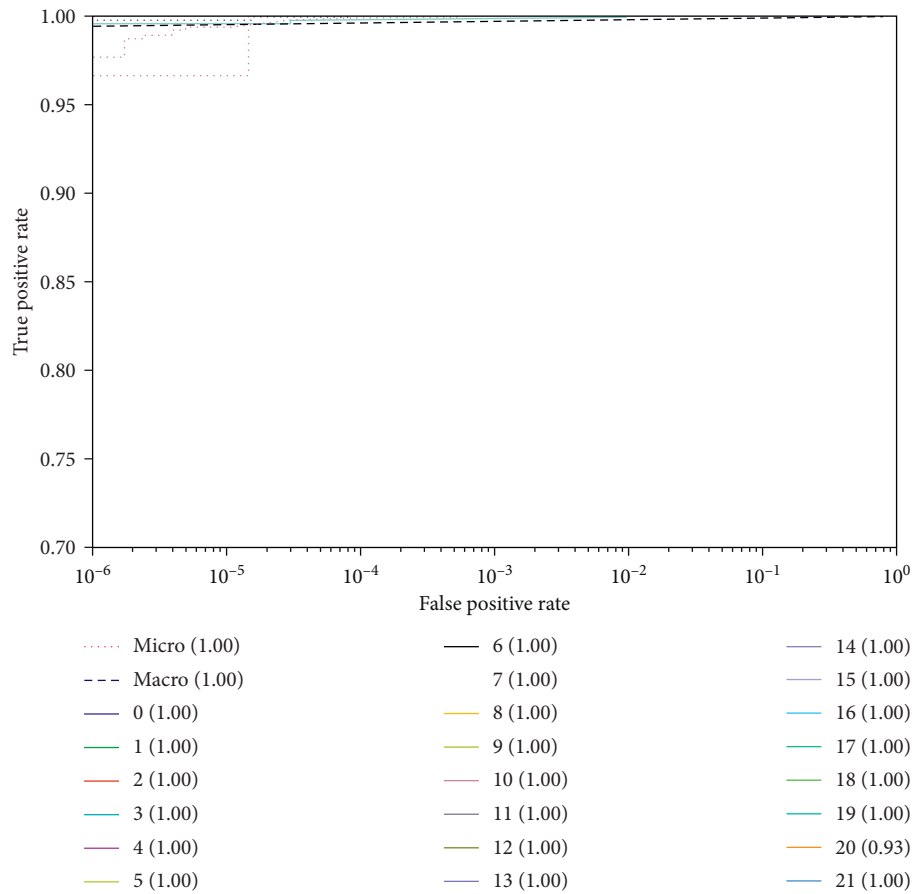


FIGURE 10: MobileNetV2 ROC curve.

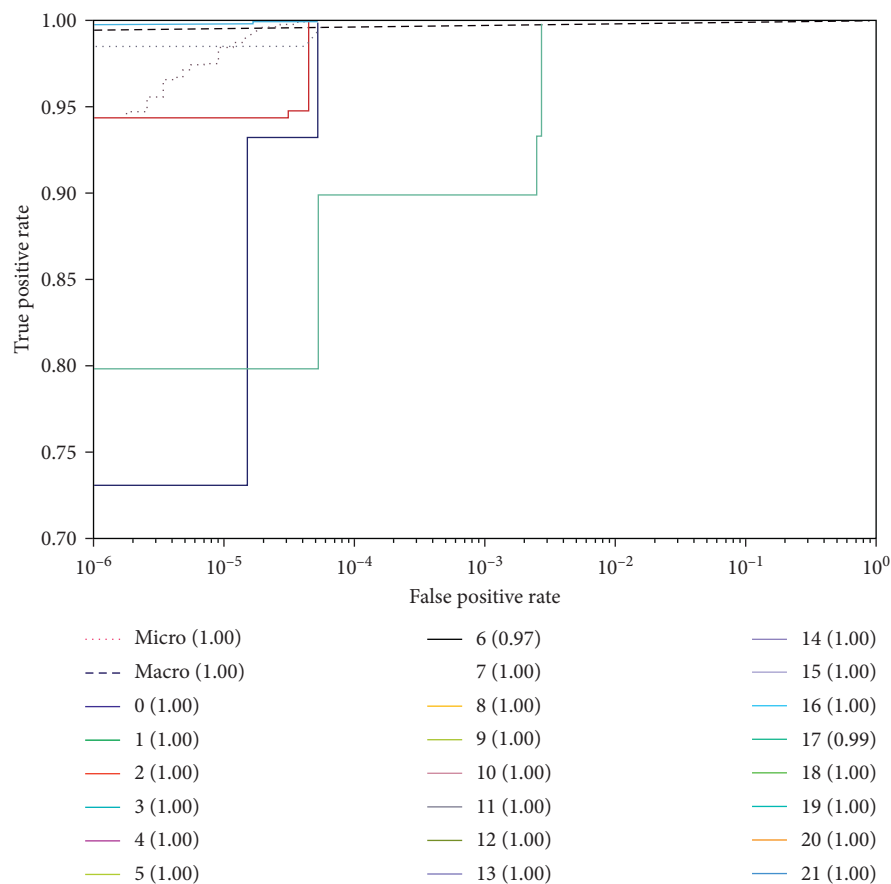


FIGURE 11: SqueezeNet ROC curve.

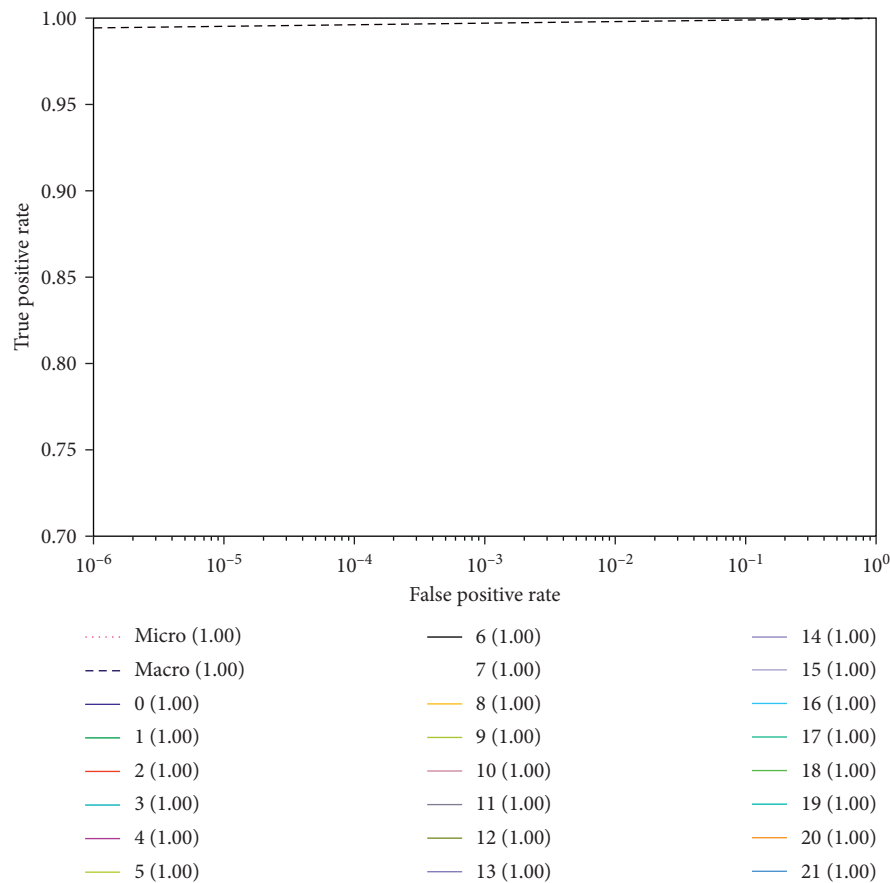


FIGURE 12: Faster SqueezeNet ROC curve.

TABLE 2: Parameters and runtime of the models.

CNN architecture	Data type	Compressed model size (MB)	TPR (FPR = $10e-6$) (%)	AVG inference time (ms)	AVG TensorRT inference time (ms)
PCA SVM	32	0.875	93.29	0.29	—
ResNetV2	32	9.10	96.995	4.65	—
DenseNet	32	2.95	96.995	3.39	—
EfficientNet	32	13.1	97.996	4.72	—
MobileNetV2	32	8.75	99.998	4.83	—
SqueezeNet	32	0.47	96.391	3.53	—
Ours (faster SqueezeNet)	32	0.255	99.999	2.67	0.65

all components are classified, how to address the many types of problems becomes very difficult. One of the solutions is to classify the main categories of components first and then divide the main categories into smaller categories. It is hoped that more researchers can participate in the research of electronic component classification.

Data Availability

The data used to support the findings of the study are available from the corresponding author via yyxu@hqu.edu.cn.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the National Key R&D Program of China (2017YFA60700602 and 2018YFC0809200) and Ministry of Industry and Information Technology of China under Project “Industrial Internet Platform Test Bed For Optimizing the Operation of Motor and Driving Equipment” (2018-06-46).

References

- [1] NASA parts selection list, <https://nepp.nasa.gov/npsl/index.htm>.
- [2] FSC standard classification of military electronic components, https://tech.hqew.com/fangan_1652961.
- [3] China Electronic Technology Standardization Institute. GB/T 17564.4-2001, *IEC Standard Data Element Types and Related Classification Models for Electrical Components-Part 4: IEC Standard Data Element Types, Component Classifications, and Reference Sets of Items*, China Standard Press, Beijing, China, 2001.
- [4] Standardization Institute of Ministry of Electronic Industry, China Electronic Statistics Society. SJ/T 11144-1997, *Electronic Product Classification and Code*, vol. 7, Ministry of Electronics Industry, PRC, Beijing, China, 1999.
- [5] B. Wang, "Base and current situation of data standardization for electronic components & devices," *Electronic Component & Device Applications*, vol. 11, pp. 30–32, 2010.
- [6] L.-x. Zheng, P.-l. Yang, and G.-j. Nie, "The research of components information platform construction and standardized processing technology," *Electronics Quality*, vol. 6, pp. 45–49, 2008.
- [7] J. Na, Y. Xing, and R. Costa-Castello, "Adaptive estimation of time-varying parameters with application to roto-magnet plant," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 7, pp. 19471–19487, 2018.
- [8] Y. Zhu, W. X. Zheng, and D. Zhou, "Quasi-synchronization of discrete-time lur'e-type switched systems with parameter mismatches and relaxed PDT constraints," *IEEE Transactions on Cybernetics*, vol. 50, no. 5, pp. 2026–2037, 2020.
- [9] J. Na, A. S. Chen, Y. A. Huang et al., "Air-fuel ratio control of spark ignition engines with unknown system dynamics estimator: theory and experiments," *IEEE Transactions on Control Systems Technology*, vol. 99, pp. 1–8, 2019.
- [10] X. Chen, J.-d. Yu, X.-a. Chen, and Y. Zhai, "Classification of electronic components based on convolution neural network," *Wireless Communication Technology*, vol. 27, no. 2, pp. 7–12, 2018.
- [11] D. Lefkaditis, G. Tsirigotis, and G. Tsirigotis, "Morphological feature selection and neural classification for electronic components," *Journal of Engineering Science and Technology Review*, vol. 2, no. 1, pp. 151–156, 2009.
- [12] Y. Zhu and W. X. Zheng, "Multiple lyapunov functions analysis approach for discrete-time-switched piecewise-affine systems under dwell-time constraints," *IEEE Transactions on Automatic Control*, vol. 65, no. 5, pp. 2177–2184, 2020.
- [13] J. Na, B. Jing, G. Gao, Y. Huang, and C. Zhang, "Unknown system dynamics estimator for motion control of nonlinear robotic systems," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 5, pp. 3850–3859, 2020.
- [14] Y. Zhu and W. Zheng, "Observer-based control for cyber-physical systems with periodic DoS attacks via a cyclic switching strategy," *IEEE Transactions on Automatic Control*, vol. 65, no. 8, pp. 3714–3721, 2020.
- [15] S.-S. Du, Z.-D. Shan, Z.-C. Huang, H.-Q. Liu, and S.-L. Liu, "The algorithmic of components auto-classification and system development of electronic components," *Development & Innovation of Machinery & Electrical Products*, vol. 6, pp. 133–135, 2008.
- [16] T. Wang, Z. Cao, S. Wang et al., "Privacy-enhanced data collection based on deep learning for internet of vehicles," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6663–6672, 2020.
- [17] Y. Xu, G. Yang, and T. Yao, "An EMD-SVM model with error compensation for short-term wind speed forecasting," *International Journal of Information Technology and Management*, vol. 18, no. 2/3, pp. 171–178, 2019.
- [18] T. Wang, H. Luo, X. Zeng, Z. Yu, A. Liu, and A. Kumar Sangaiah, "Mobility based trust evaluation for heterogeneous electric vehicles network in smart cities," *IEEE Transactions on Intelligent Transportation Systems*, vol. 99, pp. 1–10, 2020.
- [19] M. Moetesum and S. W. Younus, M. A. Warsi and I. Siddiqi, Segmentation and recognition of electronic components in hand-drawn circuit diagrams," *ICST Transactions on Scalable Information Systems*, vol. 5, no. 16, Article ID 154478, 2018.
- [20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, no. 2, pp. 84–90, 2017.
- [21] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: surpassing human-level performance on ImageNet classification," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, IEEE, Santiago, Chile, December 2015.
- [22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, pp. 1068–1082, San Diego, CA, USA, May 2015.
- [23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [24] M. Kalash, M. Rochan, N. Mohammed, and N. D. B. Bruce, "Malware classification with deep convolutional neural networks," in *Proceedings of the 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, IEEE, Paris, France, February 2018.
- [25] B. Du and L. Zhang, "Target detection based on a dynamic subspace," *Pattern Recognition*, vol. 47, no. 1, pp. 344–358, 2014.
- [26] B. Du, Z. Wang, L. Zhang et al., "Exploring representativeness and informativeness for active learning," *IEEE Transactions on Cybernetics*, vol. 47, no. 1, pp. 14–26, 2015.
- [27] C. Huang, Y. Li, C. C. Loy, and X. Tang, "Learning deep representation for imbalanced classification," in *Proceedings of the Computer Vision & Pattern Recognition*, IEEE, Las Vegas, NV, USA, June 2016.
- [28] N. Japkowicz and S. Stephen, "The class imbalance problem: a systematic study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–449, 2002.
- [29] J. Fan, X. Cao, P.-T. Yap, and D. Shen, "BIRNet: brain image registration using dual-supervised fully convolutional networks," *Medical Image Analysis*, vol. 54, pp. 193–206, 2019.
- [30] M. Tan and Q. V. Le, "EfficientNet: rethinking model scaling for convolutional neural networks," 2019, <https://arxiv.org/abs/1905.11946>.
- [31] Y. Xu, W. Yan, H. San, G. Yang, and J. Luo, "CenterFace: joint face detection and alignment using face as point," *Scientific Programming*, vol. 2020, Article ID 7845384, 8 pages, 2020.
- [32] T. Wang, L. Qiu, A. K. Sangaiah, A. Liu, M. Z. A. Bhuiyan, and Y. Ma, "Edge-Computing-based trustworthy data collection model in the internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4218–4227, 2020.
- [33] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural networks," 2015, <http://arxiv.org/abs/1506.02626>.

- [34] J. Hu, Li Shen, S. Albanieand, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2017.
- [35] J. Pankaj, R. Gaurav, D. Debabrata, and M. Siddhartha, "Automatic sleep stage classification using time–frequency images of CWT and transfer learning using convolution neural network," *Biocybernetics and Biomedical Engineering*, vol. 40, no. 1, 2020.
- [36] A. G. Howard, M. Zhu, B. Chen et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," 2017, <http://www.arxiv.org/abs/1704.04861>.
- [37] A. Howard, M. Sandler, G. Chu et al., "Searching for MobileNetV3," 2019, <https://arxiv.org/abs/1905.02244>.
- [38] G. Huang, Z. Liu, L. Van Der Maaten, and K. Weinberger, "Densely connected convolutional networks," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.
- [39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2016.
- [40] F. N. Iandola, S. Han, M. W. Moskewicz et al., "SqueezeNet: alexnet-level accuracy with 50x fewer parameters and <0.5 mb model size," 2016, <https://arxiv.org/abs/1602.07360>.
- [41] X. Liu, P. Lin, L. Tang, T. Wang, A. Liu, and W. Xu, "Objective-variable tour planning for mobile data collection in partitioned sensor networks," *IEEE Transactions on Mobile Computing*, vol. 99, 2020.
- [42] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," 2016, <https://arxiv.org/abs/1608.06993>.
- [43] C. Szegedy, S. Ioffe, V. Vincent, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," 2016, <https://arxiv.org/abs/1602.07261>.
- [44] NVIDIA TensorRT: NVIDIA Developer, <https://developer.nvidia.com/tensorrt>.