RESEARCH ARTICLE

Research on Volume Measurement of Logs Based on Embedded Application

JISHI ZHENG[1], SHAOYI LI [2], SHIWEN ZHANG [2], LINGHUA KONG[3], AND DING ZHIGANG[3]
[1]Intelligent Transportation System Research Center, Fujian University of Technology, Fuzhou 350118, China
[2]School of Transportation, Fujian University of Technology, Fuzhou 350118, China
[3]School of Mechanical and Automotive Engineering, Fujian University of Technology, Fuzhou 350118, China

Corresponding author: Shaoyi Li (2639143384@qq.com)

ABSTRACT Spurred by the worldwide concern for forest protection and the increased log sales, most countries have standardized log volume calculations to avoid excessive timber and protect buyers. However, log volume is currently manually measured, suffering from high labor costs, low measurement progress, and imposing significant measurement errors. Thus, automatically obtaining the volumetric data of logs is a convenient and quick solution. Therefore, this work proposes a Mask Region Convolutional Neural Network-based (R-CNN) algorithm for Logs volume measurement, named Wood Mask (WM) R-CNN. Specifically, we employ the Res2Net structure as the backbone to obtain receptive fields that exceed the input feature size, thus improving our model's multi-scale information fusion ability. Additionally, WM R-CNN relies on the Path Aggregation Feature Pyramid Network's (PAFPN's) path enhancement structure, shortening the low-level feature map's propagation path and improving the wood contour segmentation accuracy. Extensive experiments on the Vehicle-mounted Dense Logs (VMDL) dataset demonstrate that WM R-CNN affords a highly appealing segmentation accuracy for small, medium, and large wood, improving the corresponding mAP indicators against current methods by 2.0%, 1.2%, and 4.4%, respectively. Furthermore, a quantitative method based on TensorRT compresses the proposed model to deploy the WM R-CNN to mobile embedded devices. However, to compensate for the quantization loss, we introduce the expansion convolution operation method to manipulate the mask map and control the volume calculation error of all logs on a vehicle within 1%. The experiments reveal that the proposed method offers an appealing performance, verifying the algorithm's effectiveness and implementation ability on mobile terminals.

INDEX TERMS Embedded device, improved mask R-CNN, logs volume measurement, TensorRT.

I. INTRODUCTION

Countries worldwide aim for carbon neutrality and control their carbon peaking, with this dual carbon issue becoming a hotspot for global social and economic development [1]. Several studies have demonstrated a positive correlation between economic growth and carbon dioxide emissions [2]. Thus reducing carbon emissions while developing the economy has become a major global challenge. Studies have revealed that forestry is an important option for emission reduction [3], as it

can help mitigate climate change by storing carbon in trees, forest soils, and wood products [4]. Hence, forest protection is mandatory to alleviate global warming, and therefore, humans must increase the protection of forests by improving the maximum rational utilization of forest resources.

A log is a piece of wood cut into a certain length according to the standard or special regulations of size, shape, and quality [5]. Currently, log sales mainly consider log volume as a measurement standard, with log processing and usage depending on the log volume measurement results. To standardize log volume, China has formed a complete set of standards to calculate Log volume (Log Volume table,

The associate editor coordinating the review of this manuscript and approving it for publication was Turgay Celik [ID].

GB/T 4814-2013), which require obtaining the inspection length and diameter of the logs and calculating their actual volume based on a formula. Therefore, measuring the log diameter is an important process, where log diameter subtle differences determine the grade of logs and affect the price and economic losses of the industry. Currently, in the production enterprises of logs [6], the monthly log diameter class is determined manually, with workers using various measuring tools, such as tape measure or caliper, to measure the length and diameter of logs in Figure 1. Then, these measurements are input into the volume calculation formula to calculate the actual volume of logs. Nevertheless, such a labor-intensive process fatigues the inspectors affecting the measurement accuracy, which cannot be guaranteed to exceed 70-80% [7], [8]. Additionally, manual measurement is subjective, time-consuming, and deviates from the latest development trend of modern log processing and sales. Opposing manual measurement techniques, automatically obtaining the volumetric data of logs is a convenient and quick solution, reducing the risk of errors during manual measurements.

FIGURE 1. Manual measurement.

Recently, computer vision and deep learning methods have progressed significantly, developing state-of-the-art solutions in various tasks such as object detection, classification, and analysis, which have been employed in several domains, including medicine, agriculture [9], autonomous driving, robotics [10], and security [11], [12]. Currently, in object detection, the You-Only-Look-Once (YOLO) object detection family of algorithms has proven its appealing detection capabilities. Specifically, the YOLOv4 [13] and YOLOv5 [14] versions are gaining significant attention from the community due to their exceptional object detection performance. Furthermore, the Transformers [15] neural network architecture, which completely relies on Attention modules and neglects Convolutional layers, achieves superior detection quality at a significantly lower computational complexity.

Existing computer vision, digital image processing, and deep learning methods have been used to identify log end faces through log images, obtain the diameter of the logs, and calculate log volume. Some scholars have made some progress in intelligent log detection, where the existing

methods can be categorized into computer vision, image processing, statistical analysis, and deep learning.

A. IMAGE PROCESSING METHODS

Yella et al. [8] utilized photos acquired by drivers and exploited color information and geometric operators in multiple color spaces to segment images and extract relevant information for automatic detection, counting, and classification of wood based on a Circular Hough Transform (CHT) algorithm. This algorithm is robust to external factors such as lighting conditions and camera differences due to mud or snow cover. Similarly, [16] combined CHT, Local Circularity Measure (LCM), and Graph Cut to estimate the weight of a wood stack. However, this method suffers from sensitivity to object distortions and noise, computational complexity, and unknown a priori object size. To solve the diameter measurement problem. Budiman et al. [17] built a simple handheld device using a fixed-length iron rod, a camera, and a raspberry PI. Specifically, they put an iron rod against the end face of the measured log and used the camera to collect images. Then the image was transferred to a Raspberry PI for compression, gray transformation, contour search, and circle fitting, and performed several operations to measure the wood diameter. This scheme has portability advantages and is easy to operate, with the measurement error controlled within 3%. However, this method requires a LED light source, as the system's robustness is insufficient under natural lighting conditions.

Moreover, Kruglov et al. [18], [19] identified the spatial structure of logs by modeling them in three dimensions and obtaining their volume. The authors combined various image processing schemes involving mean-shift clustering, Delaunay triangulation, Boruvka's minimum spanning tree algorithm, watershed algorithm, and Boykov-Kolmogorov graph cut algorithm. Although this method attains some appealing results, the computational complexity is high and is affected by lighting conditions. In [20], the authors introduced an improved fast radial symmetry algorithm for log detection. The watershed algorithm and Stoer-Wagner algorithm were applied on the identified log edges to discard wrong targets and realize wood counting and segmentation. Compared with manual measurement, the average final log volume calculation error is 9.2%, and the error rate of volume detection is higher.

B. COMPUTER VISION METHODS

Besides, [21] automatically detected logs by combining the Histogram of Gradients (HoG) features with a Random Decision Forest scheme. However, the log shape is constrained to highly round-shaped logs, and the HoG features are significantly affected by illumination conditions. In [22], the authors developed a multi-view Structure from a Motion (SfM) photogrammetry scheme to create 3D models for automated volumetric measurement of truckloads. The log images are acquired using a small UAV flying around logging trucks transporting Eucalyptus nitens pulplogs. Although this

method presented a low error of approximately 3%, it is complex, requires a flying UAV, and is affected by illumination conditions. An ensemble of weak classifiers comprising rectangular Haar features has also been proposed [23].

C. STATISTICAL ANALYSIS METHODS

The literature also suggested an iterative classification and segmentation strategy that relied on Gaussian mixture models [24]. However, this method is limited to log detection and does not afford log volume estimation.

D. DEEP LEARNING

Spurred by the advantages of deep learning, Tang et al. [25] detected logs in natural scenes based on the Single Shot MultiBox Detector (SSMD) deep learning model. Specifically, this scheme employs the annotation information of the log face image to learn the features of the log face area, reduces the background interference during target recognition, and enhances the model's learning ability. This method's accuracy is 94.87%, and the recall rate is 91.34%. Compared with traditional methods, this strategy reduces the influence of ambient light on the log recognition rate, solves the log face detection and recognition problem under complex background conditions, and lays a foundation for intelligent volume calculation of logs.

Furthermore, Samdangdech et al. [26] introduced a deep learning architecture that combines the SSD model with a Fully Convolutional Network (FCN) semantic segmentation model for vehicle eucalyptus image segmentation. The SSD model detects the location of logs on the truck, extracts the log area, and then the FCN network completes the segmentation of logs. Finally, the segmentation mask map output by the FCN model undergoes morphology processing, and the logs are counted. The experimental results reveal that the correct wood segmentation rate is 94.45%, but some logs with no obvious features are not properly segmented, and the missing segmentation phenomenon is severe.

Table 1 summarizes the existing works on log detection, highlighting their capabilities and limitations. Although the existing literature suggests several methods for log detection and volume estimation, these present the following deficiencies.

a. Poor log recognition, as existing methods are affected by lighting conditions, noise, distortions, and clutter objects posing a high error rate and thus demonstrating poor robustness.

b. Require a controlled lighting environment to overcome their influence on lighting conditions, posing an additional constraint on cost and hardware complexity setup.

c. Few methods can calculate the log volume, while most only conduct log detection.

d. The influence of the shooting distance between the target and the camera on the final log end diameter estimation is not considered.

e. Existing deep learning-based methods that calculate log volume focus on the algorithm's recognition effect without focusing on execution efficiency.

f. To our knowledge, there is no mature model for actual deployment and application.

To overcome these problems, this paper proposes a deep neural network-based scheme for log volume estimation (overcoming limitation (c) of the existing methods), which relies on the Mask Region Convolutional Neural Network (R-CNN). However, in this work, Mask R-CNN has been improved by employing the Res2Net module and the Path Aggregation Feature Pyramid Network (PAFPN) structure to enhance the detection performance of logs. Indeed, the developed architecture is robust to varying lighting conditions and other external distortions, overcoming limitations (a) and (b) of the current methods. Considering the actual stratified detection requirements, we suggest an adaptive stratified log diameter estimation method that combines K-means and depth information, solving the limitation (d) of current log volume estimation methods. Finally, regarding execution efficiency, we quantize and compress the model by adopting the existing deep learning quantization strategies (overcoming limitation (e)). The recognition accuracy of the quantized model is challenged on float32, float16, and int8 data types while deployed on an embedded edge device (overcoming limitation (f)).

The remainder of this article is organized as follows. Section II briefly introduces the basic structure of the Mask R-CNN network model and describes improvements to the original Mask R-CNN network model. Section III mainly introduces the deployment strategy of the model, the fitting method of the target mask, and the concrete implementation of the self-adaptive stratified log diameter estimation method combining K-means and depth information. Section IV mainly introduces the source of experimental data sets, the configuration of model training parameters, and evaluation indicators. Section V further presents the results and discussion. Finally, the paper summarizes and proposes the research directions. Table 2 summarizes the notations used.

II. MATERIALS AND METHODS
A. DATASET

This work aims to achieve independent segmentation of the wood contour, so the team using the polygonal annotation method can effectively label the boundary between the bark and tree core of each wood within the image and label the wood category as ''wood''. We consider pictures of logs loaded on vehicles, typically between 50 and 300, with the inspection ruler diameter randomly ranging from 4 cm to 40-50 cm. We use industrial and mobile phone cameras to collect 150 images of logs, including clear images under different lighting conditions, wood end backgrounds, and shooting angles. All images are labeled using the Labelme image annotation tool, with the annotation effect illustrated in Figure 2. This data set is named the Vehicle-mounted Dense Logs (VDL). The labeled images are divided into training,

TABLE 1. Log measurement methods.

method	category	operating principle	capabilities	limitations
Yella et al. [8]	computer vision	CHT	detection, counting, and classification of wood	no log diameter or volume
Galsgaard et al. [16]	computer vision	combining CHT, LCM, and Graph Cuts	log weight estimation	sensitivity to object distortions and noise, computational complexity, and unknown a priori object size
Budiman et al. [17]	image processing	various image processing techniques	measures wood diameter	requires LED lighting, no log volume
Kruglov et al. [18, 19]	image processing	various image processing techniques	3D modeling to obtain the volume	high computational complexity, affected by lighting conditions
Kruglov et al. [20]	image processing	fast radial symmetry, watershed algorithm, Stoer-Wagner algorithm	log volume estimation	high log volume error
Chiryshev et al. [21]	computer vision	HoG and Random forests	log detection	no log volume, affected by lighting conditions, the log must be round
Acuna and Sosa [22]	computer vision	3D reconstruction using Structure from Motion	log volume calculation, low error	complex method requires UAV, affected by illumination conditions
Gutzeit and Voskamp [23]	computer vision	Ensemble of Haar features	log detection	no log volume
Herbon et al. [24]	statistical analysis	GMM	log detection	no log volume
Tang et al. [25]	deep learning	SSMD model uses annotation information of the log face image to learn the log face features	ambient light does not affect the log recognition rate, can detection and recognize log faces under complex background conditions, appealing accuracy, and recall	distorted contour
Samdangdech et al. [26]	deep learning	combining SSD and FCN	high wood segmentation rate	segmentation deadens on wood features, severe missing segmentation phenomenon
proposed	deep learning	combining Mask R-CNN and PAPFN	-	-

validation, and test sets according to a 4:1:1 ratio [27](see Table 3).

A common method to improve a model's generalization ability is to expand the data set through data augmentation. Here, we apply horizontal flipping, enhance the image's red, blue, and green channels, enhance the image's pixel values, apply Gamma correction, salt and pepper noise, contrast enhancement, sharpening, and blurring, and cut and scale the training images. Finally, the enhanced dataset is increased by 16 times.

B. IMPROVED MASK R-CNN NETWORK

1) COLOR/GRAYSCALE FIGURES

Kaiming et al. [28] extended the Faster R-CNN [29] target detection model and introduced Mask R-CNN. By adding the mask segmentation network, Mask R-CNN achieves pixel-level independent segmentation of targets belonging to the same category. The Mask R-CNN structure is illustrated in Figure 3 and mainly comprises the Backbone, Region Proposal Network (RPN), RoI Align, and R-CNN network.

The backbone network comprises ResNet [30] and FPN (Figure 4). Specifically, the log image is input into the model, with ResNet extracting the log contour features and outputs five feature maps of different scales C1, C2, C3, C4, and C5. Then, the feature pyramid establishes connections between the five feature maps with different scales and outputs the effective feature layers P2, P3, P4, P5, and P6 after feature fusion. The effective feature layer obtained here has two applications, as it can be directly used in combination with the ROI Align layer, and it is input into the RPN network
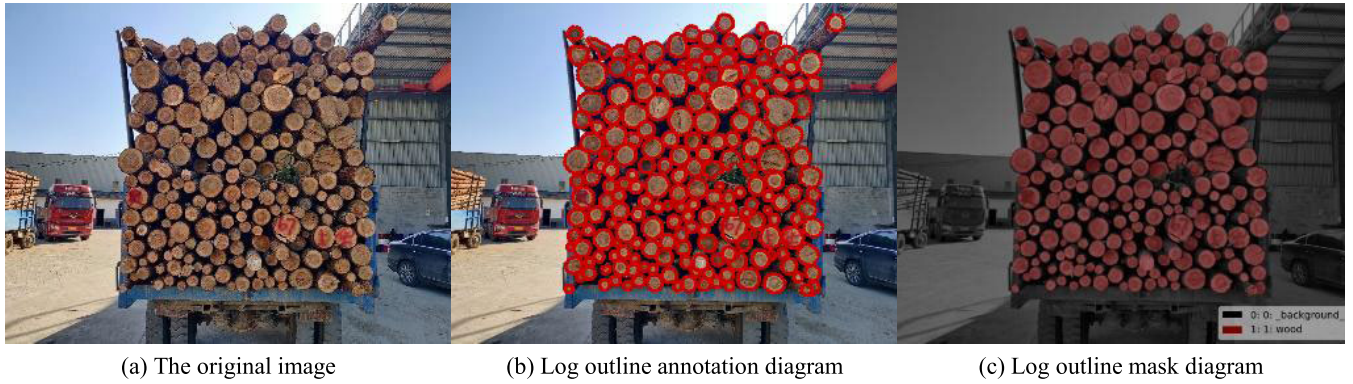
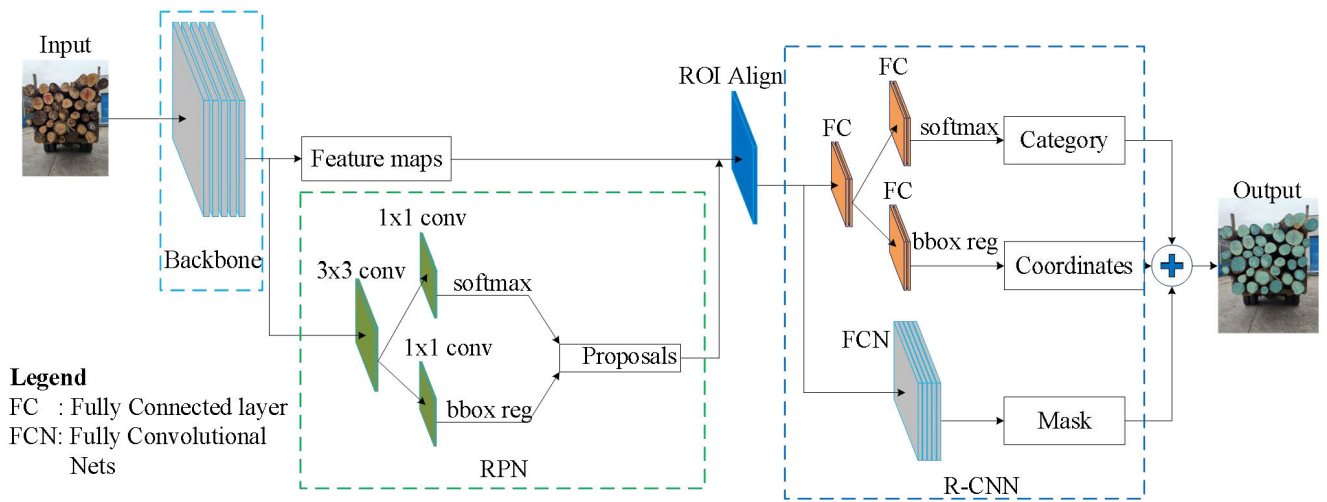| (a) The original image | (b) Log outline annotation diagram | (c) Log outline mask diagram |

FIGURE 2. Log outline annotation effect.

FIGURE 3. Structure diagram of Mask R-CNN model (FC denotes the fully connected layer).

TABLE 2. List of symbols.

symbol	description
x_n	image's feature subset
$K_n()$	convolution kernel
(x_c, y_c)	elliptic origin coordinates
θ_{offset}	elliptic deflection Angle
l_{AB}	logs' face minor axis
a	major axis of ellipse
b	minor axis of ellipse
z	the distance from the center of the log to the camera
$A(x_1, y_1)$	terminal coordinates of the minor axis of the ellipse
$B(x_2, y_2)$	terminal coordinates of the minor axis of the ellipse
f_x , f_y	camera inside and outside parameters

TABLE 3. Annotate the image dataset log statistics table.

Dataset Information	All	Small	Medium	Large
Dataset	15533	3432	11753	348
Train	10743	2522	7949	272
Validation	2391	468	1882	41
Test	2399	442	1922	35

After that, the proposed network has two branches: the upper is responsible for classification and regression, and the lower for generating the corresponding mask.

2) IMPROVED RESNET NETWORK

In order to improve the ability of network feature extraction and the detection performance of large and small woods, this paper introduces Res2Net [19] module to replace the Bottleneck residual block in the original Mask R-CNN [16] backbone network. The original Bottleneck residual block and Res2Net [19] structure block are shown in Figure 5.

to generate a target candidate box by using a non-maximum suppression algorithm. Then the ROI Align layer will use these candidate boxes to intercept the effective feature layer.
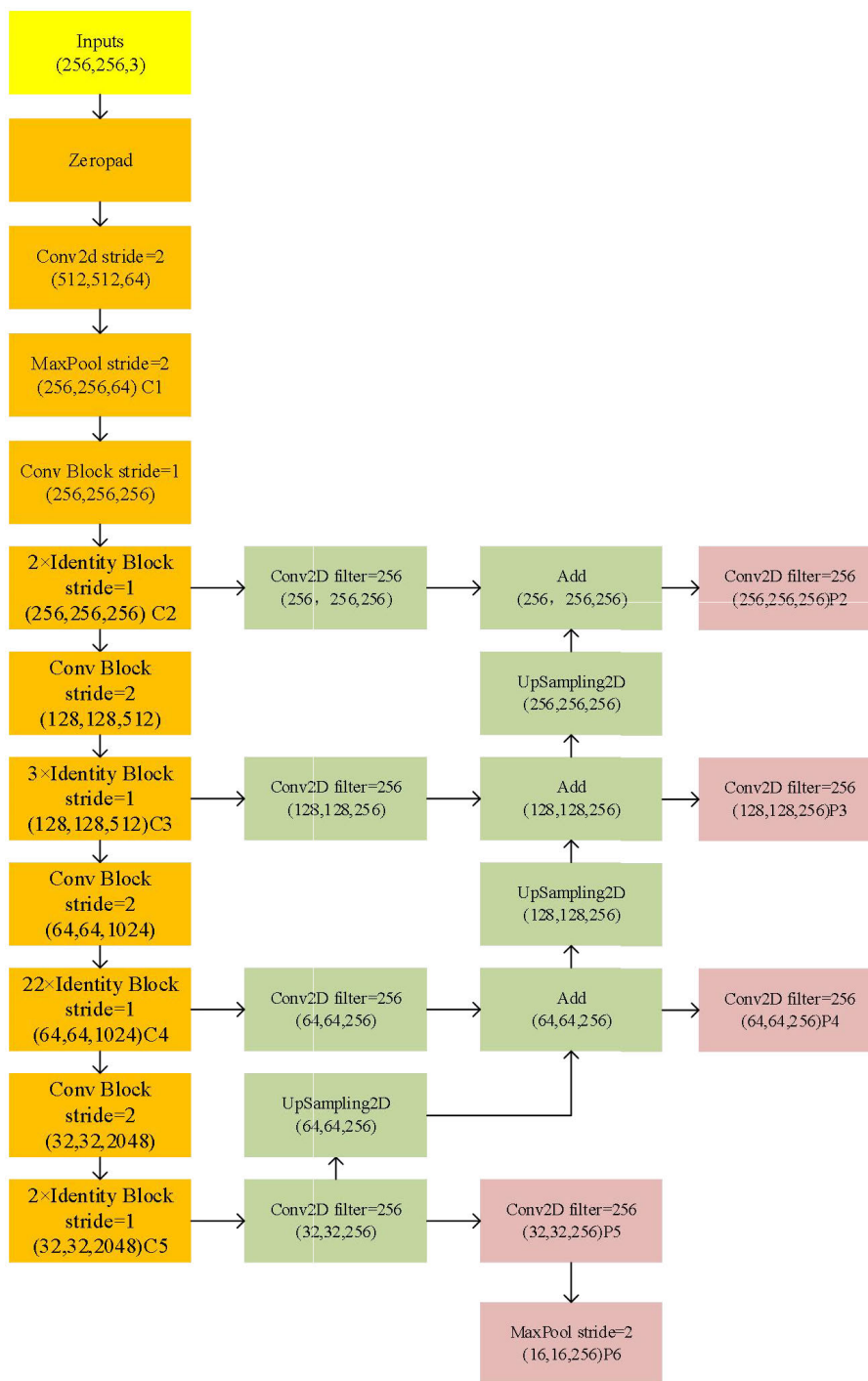
FIGURE 4. Schematic diagram of backbone network structure.

Compared with the original Bottleneck residuals, the Res2Net block introduces a new dimension called ''scale'', which modifies the original 3*3 convolution kernel of the residual block. Besides, after the 1*1 convolution, the feature map is no longer a 3*3 convolution but is replaced with a more complex structure. Specifically, the feature map of n channels is evenly divided into s feature subsets after a 1*1 convolution with a step size of 1 denoted by x_n, where $n = [1, s]$ and the number of channels per feature subset is n/s. The direct output of x_1 is y_1, the output of the x_2 feature subset is y_2 after connecting a 3*3 convolution kernel $K_n()$, and the subsequent feature subset x_n is first added with the convolution result of the previous feature subset, and then the output is y_n after a 3*3 convolution. The mathematical
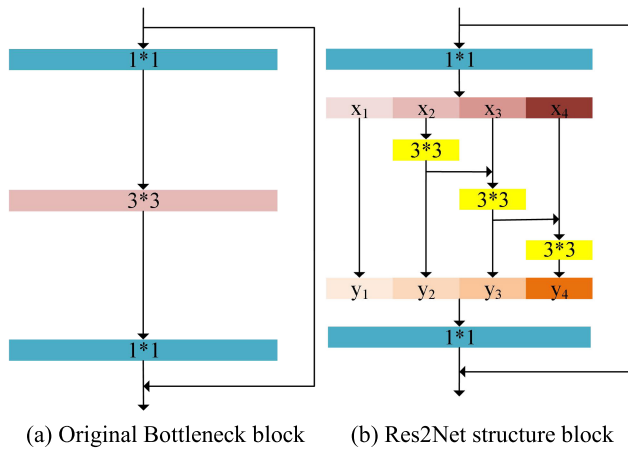
(a) Original Bottleneck block   (b) Res2Net structure block

FIGURE 5. Comparison of Res2Net and bottleneck block.

expression of the Res2Net structural block operation process is [31]:

$$y_n = \begin{cases} x_n, & i = 1; \\ K_n(x_n), & i = 2; \\ K_n(x_n + y_{n-1}), & 2 < n \leq s. \end{cases} \quad (1)$$

Finally, the output result of each feature subset is concatenated along the channels and then added to the input feature map after a 1*1 convolution. In this way, Res2Net involves receptive fields that exceed the input feature size by performing scale segmentation on the feature map channel dimension and then conducting several operations on each small feature map to improve the model's multi-scale information fusion ability.

3) IMPROVED FPN NETWORK

The FPN [32] structure in the Mask R-CNN network is associated with the C2-C5 feature map output after the ResNet network feature extraction process, and the information fusion among each feature map is realized from top to bottom. However, ResNet is a bottom-up structure, opposing the FPN's top-down structure [33], and thus increases the information path of the low-level feature map when propagating toward the high-level and reducing the target's positioning accuracy and segmentation accuracy. To solve the problem of low-level information Path propagation, this paper replaces the original FPN network with the Path Aggregation Feature Pyramid Network (PAFPN) [34]. Figure 6 illustrates the PAFPN network.

Considering N2 = P2, from N3 to N5, each feature map Ni is first compressed by a 3*3 convolution with step size 2 and then added to the feature map Pi+1 output by the FPN to undergo a feature fusion process. Then, another 3*3 convolution with a step size of 1 is used to process the feature map output Ni+1 after the feature fusion process. Based on N5, we obtain N6 by connecting a maximum pooling layer with a step size of 2. In this way, the original FPN network output feature graph P2, P3, P4, P5, and P6 is replaced by N2, N3,

N4, N5, and N6, respectively. Thus, the PAFPN path, which has a reinforced structure for its information fusion layers, significantly reduces the lower figure propagation paths to the top, preventing low-level information from traveling backward and thus neglecting the multiple convolution layer loss in ResNet due to the target location information. This strategy further improves the wood contour segmentation accuracy and the quality of the output mask.

C. POST-PROCESSING

1) THE FITTING METHOD OF LOG END FACE ELLIPSE BASED ON THE LEAST SQUARE METHOD

Compared with circle fitting, the elliptical fitting of a log contour is more in line with the log diameter measurement standard described in the Log volume table GB/T 4814-2013. Thus, to obtain the inspection ruler diameter of the log end face, this paper adopts the Least Square Method(LSM) ellipse fitting method to fit the log contour.

The LSM is a commonly used mathematical optimization algorithm that minimizes the sum of squares of ellipse fitting errors by optimizing the objective function to find the best match between the input target contour and the ellipse contour [35]. For any ellipse, its inhomogeneous equation is:

$$x^2 + Axy + By^2 + Cx + Dy + E = 0 \quad (2)$$

In this case, the coordinates of the center of the ellipse can be

$$\begin{cases} x_c = \dfrac{2BC - AD}{A^2 - 4B} \\ y_c = \dfrac{2D - AC}{A^2 - 4B} \end{cases} \quad (3)$$

In this case, the deflection Angle θ_{offset} is:

$$\theta_{offset} = -\frac{1}{2}\arctan\left(\frac{A}{B-1}\right) \quad (4)$$

The long and short axes can be expressed as:

$$\begin{cases} a = \sqrt{\dfrac{x_c^2 + Ax_cy_c + By_c^2 - E}{\cos^2\theta_{offfset} - A\sin\theta_{offset}\cos\theta_{offset} + B\sin^2\theta_{offset}}} \\ b = \sqrt{\dfrac{x_c^2 + Ax_cy_c + By_c^2 - E}{\sin^2\theta_{offset} + A\sin\theta_{offset}\cos\theta_{offset} + B\cos^2\theta_{offset}}} \end{cases} \quad (5)$$

A series of points $P_i(x_i, y_i)$ can be obtained by identifying the log contour from the mask diagram obtained by model inference. According to the principle of the least square method, the objective fitting function is assumed to be:

$$F(A, B, C, D, E)$$
$$= \sum_{i=1}^{n} (x_i^2 + x_iy_iA + y_i^2B + x_iC + y_iD + E)^2 \quad (6)$$

In order to find the best match between the ellipse and the target contour, F should be minimized. That is, the partial derivatives of F are 0:

$$\frac{\partial F}{\partial A} = \frac{\partial F}{\partial B} = \frac{\partial F}{\partial C} = \frac{\partial F}{\partial D} = \frac{\partial F}{\partial E} = 0 \quad (7)$$
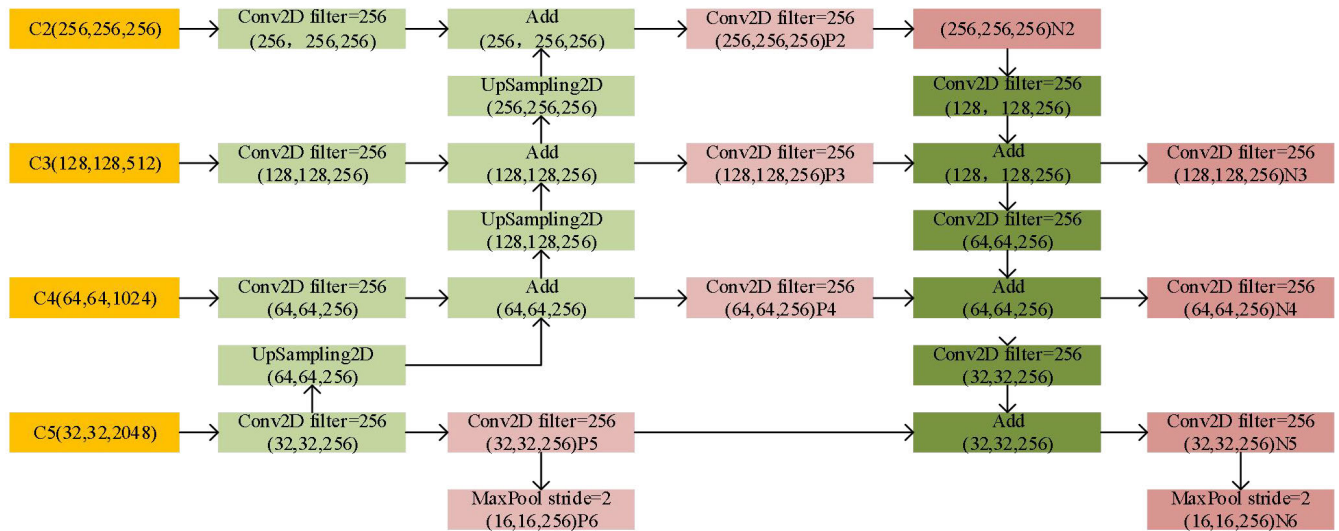
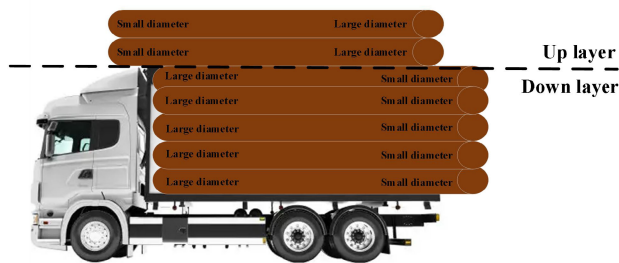**FIGURE 6.** Schematic diagram of PAFPN network.

FIGURE 7. Diagram of a car log layout in a real scene.

After solving *A*, B, C, D, E, according to (3), (4), and (5), the relevant parameters of the ellipse can be obtained, namely the center coordinate (x_c, y_c), deflection Angle θ_{offset}, major axis *a*, and minor axis *b*.

2) AN ADAPTIVE STRATIFIED LOG DIAMETER ESTIMATION METHOD BASED ON K-MEANS AND DEPTH INFORMATION

We obtain the fitting ellipse parameters of the log end mask profiles using the least squares method, including the coordinates of the ellipse center point (x_c, y_c), major axis *a*, minor axis *b*, and rotation angle θ_{offset}. However, these parameters are given as image pixels, but we require the actual log short diameter length when checking the log ruler. Nevertheless, converting pixel parameters to actual dimensions is an open problem. Moreover, considering the real application scenario, during log transportation and due to driving safety and stability factors, the loading of logs should be divided into two layers according to the specification requirements. The latter suggests that the head of the lower logs faces inside, the small head face outside, and the upper and the lower logs are in the opposite direction, as illustrated in Figure 7.

We use a ZED 2 binocular camera to acquire the log images for the experiments. By calling the camera's API, we acquire

the photos, obtain the depth images, and read the inside and outside parameters from the camera: f_x and f_y. The center point coordinates of the ellipse fitted to each log end face are the ones presented in Section I, and the distance *z* from the center point of each log to the camera was obtained by combining the depth image. Two endpoints $A(x_1, y_1)$ and $B(x_2, y_2)$ on the minor axis of the fitting ellipse were taken, and their coordinates were calculated based on the center coordinate (x_c, y_c), deflection Angle θ_{offset}, major axis *a*, and minor axis *b*. The corresponding formula is:

$$\begin{cases} x_1 = x_0 - \frac{1}{2} \cdot b \cdot \sin\theta_{offset} \\ y_1 = y_0 - \frac{1}{2} \cdot b \cdot cos\theta_{offset} \end{cases} \quad (8)$$

$$\begin{cases} x_2 = x_0 + \frac{1}{2} \cdot b \cdot \sin\theta_{offset} \\ y_2 = y_0 + \frac{1}{2} \cdot b \cdot cos\theta_{offset} \end{cases} \quad (9)$$

The actual length ofthe los' face minor axisis:

$$l_{AB} = z * \sqrt{\frac{(x_1 - x_2)^2}{f_x^2} + \frac{(y_1 - y_2)^2}{f_y^2}} \quad (10)$$

Due to the actual loading specifications, the upper and lower layers are not divided, and thus a conventional hierarchical idea measures the distance between the upper and lower layers and then manually sets the threshold. When the distance between the logs is less than this threshold, we consider the logs belonging to the lower logs and calculate the diameter of the lower logs class. This method required manual measurement and threshold setting, which is relatively tedious.

To reduce manual interference, this paper proposes an automatic stratification method based on K-means further

to improve the convenience of the onboard log measurement scale. The K-means algorithm means that the clustering involves k clusters, and the mean value of the data in each cluster is taken as the cluster's center. The K-means algorithm operates as follows: First, k samples are randomly selected from the sample set as the cluster center, and the distance between all samples and these k "cluster centers" is calculated. Each sample is divided into a cluster where the nearest "cluster center" is located. This paper takes the depth information corresponding to the center points of all logs identified as the sample set. For the task to be solved, as long as logs are divided into upper and lower layers, the number of clusters is K=2. Once the K-means algorithm completes its calculations on the sample set, two cluster centers are obtained, where one belongs to the upper log and the other to the lower log. The average of the two clustering centers is the threshold for stratification, and then each log is traversed to judge the depth information of the log. When the depth information is less than the threshold, the log is considered to belong to the lower layer, and then its diameter can be estimated. The stratification algorithm is presented in Figure 8.

D. QUANTITATIVE DEPLOYMENT OF NETWORK MODELS

Quantization refers to the process of transforming continuous values into multiple discrete values. In deep learning, model quantization is a model compression technique that transforms the model parameters' floating point storage (operation) form into the shaping storage (operation) form. Existing deep learning frameworks often use FP32 data to represent weights and bias values. Meanwhile, many neural networks often use the deepening method to increase the depth of the convolution neural networks and improve their performance. Nevertheless, increasing the network's size imposes great challenges to deploying the models on edge computing platforms with strict power and computational requirements. Thus, reducing the power and delay of model inference becomes a key issue. Therefore, it is mandatory to make the model lightweight, speed up the reasoning speed, and reduce the model's power consumption, with model quantization being one of the most simple and effective ways to meet these requirements.

TensorRT is a high-performance deep learning model inference application optimizer that provides high throughput and low latency deployment inference for deep learning model applications on embedded platforms. Combining TensorRT and NVIDIA GPUs enables fast and efficient deployment reasoning in almost all frameworks, with TensorRT being the main tool for model quantization inference deployment in this paper. The quantization process of TensorRT is presented in Figure 9 and operates as follows:.(1) The activation value and weight are converted from FP32 to INT8 by linear mapping. (2) The convolutional layer obtains the INT32-bit activation value. If the INT8 is directly used to save, excessive cumulative loss will be caused. (3) It is converted back to INT8 as the input of the next layer by
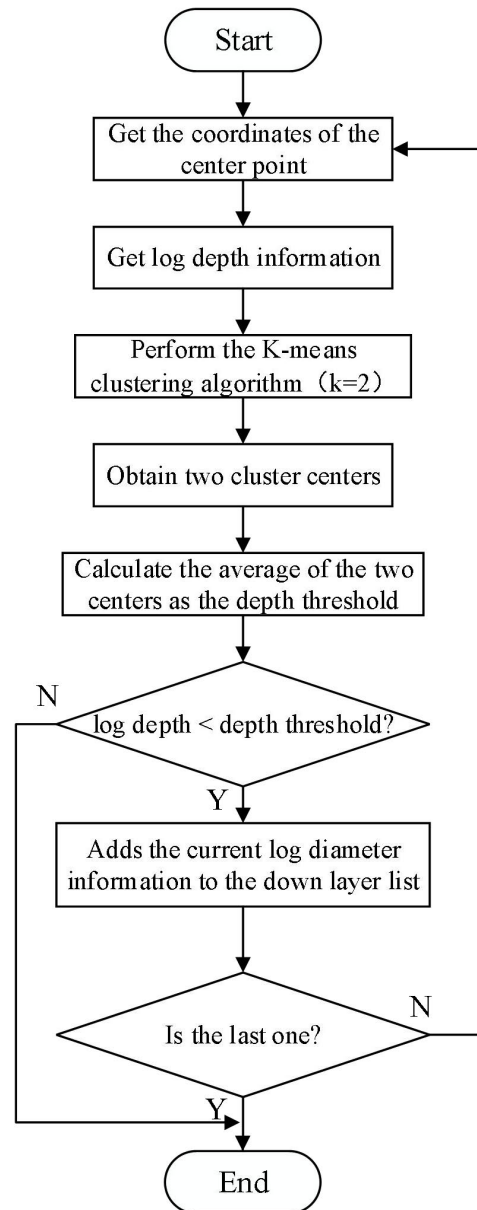

FIGURE 8. Flow chart of log stratification algorithm based on K-means.

re-quantization. (4) When the network operates at its last layer, reverse quantization converts back to FP32.

III. RESULTS AND ANALYSIS

A. TRAINING ENVIRONMENT AND HYPERPARAMETER SETUP

This paper uses GPU to speed up the network's training. Then the trained network model is deployed on Nvidia's embedded mobile device Xavier NX. Table 4 lists the experimental and platforms' hardware setup.

Specifically, we adapt MMDetection [36] to train the network model. For thetraining process, we utilize the initialization parameters ofthe pre-training weights of the ImageNet public dataset. At the same time, theother hyperparameters
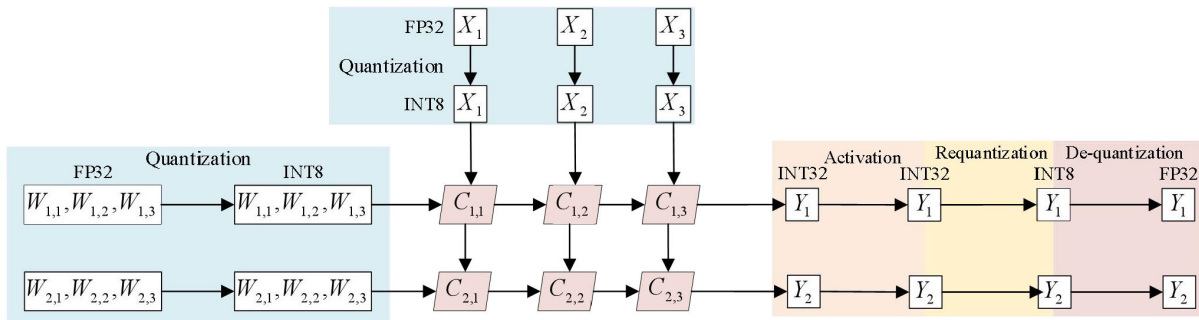
FIGURE 9. Schematic diagram of the TensorRT quantization process.

TABLE 4. Hardware platform configuration.

Platform	System	Memory	CPU	GPU
Train-platform	Ubuntu	16G	Intel i9-10850K	RTX 3090
Validation-platform	Windows	16G	Intel i7-11700K	RTX 3060
Deploy-platform	Ubuntu	8G	NVIDIA Carmel ARM®v8.2 64-bit	NVIDIA Volta™GPU

TABLE 5. Backbone network improvement and original model comparison experiment configuration.

Group	ResNet50	ResNet101	Res2Net101	FPN	PAFPN
1	√			√	
2	√				√
3		√		√	
4			√		√

are configured as follows: thenumber of detection categories is set to 1, the detection category is ''wood'', and the Rectified Linear Unit (ReLU) [37] is the activation function. Moreover, the BatchSize is set to 1, the number of data loading threads is 2, we consider 36 epochs, the gradient optimization method is the Stochastic Gradient Descend(SGD) [27], with an initial learning rate of 0.01, and the learning rate changes linearly from $lr/3$ to lr in the first 500 training iterations to stabilize the parameter gradient at the beginning of the training.

In addition to these general parameters and based on our previous work [38], the optimal parameter configuration is adopted, i.e., the RPN in the Mask R-CNN network and the positive sample sampling number of the R-CNN module are set to 512 and 1024. Moreover, the multi-scale training method is used, and the parameters are configured separately for the improved backbone network and the competitor models. Among them, the Backbone parameters of the original Mask R-CNN model include ResNet and FPN, and the improved Backbone model parameters include Res2Net and PAFPN(Table 5).

It should be noted that due to property concerns, the dataset employed and the developed algorithm are not yet publicly available. However, reproducing the code based on our paper is possible, and our code can be applied to a wide range of log datasets beyond the one presented in this paper.

B. MODEL IMPROVEMENT EXPERIMENT AND RESULT ANALYSIS

The image-enhanced wood data sets were used to complete the Backbone network improvement comparison experiment. Once the model was successfully trained on the model's training universal parameters described in Section III-A, we calculated the competitor's mean Average Precision(mAP) index on the test dataset. Additionally, the average IoU score of the mask, the number of model parameters of each experimental group, and the corresponding inference speed are reported in Table 6.

In Table 6, group 1 was taken as the control group and the other groups as the experimental groups. By comparing groups 1 and 2, we found that:When the fixed feature extraction network was ResNet50 and the feature extraction module was replaced from FPN to PAFPN, the PAFPN structure was relative to the FPN structure, although the mAP_S index decreased by 0.3%. However, the $mAP_{0.5:0.95}$, mAP_M, mAP_L, and $mIoU_{mask}$ increased by 0.3%, 0.2%, 0.4%, and 0.21%, respectively. The comparison experiment between PAFPN and FPN verified the ability of PAFPN to strengthen feature fusion and improve the performance of large object detection. By comparing groups 1 and 3, while preserving the feature extraction module unchanged, the feature extraction network deepens from ResNet50 to ResNet101 for $mAP_{0.5:0.95}$. We observed that the mAP index of a small, medium, and the large target is improved by 0.2%, 0.7%, and 3.1%, respectively. At the same time, the mask segmentation quality improved by 0.62%. By deepening the neural network, the feature extraction ability of the network can be further improved, and the higher the model's accuracy,

(a) ResNet50+FPN

(b) ResNet50+PAFPN

(c) ResNet101+FPN

(d) Res2Net101+PAFPN
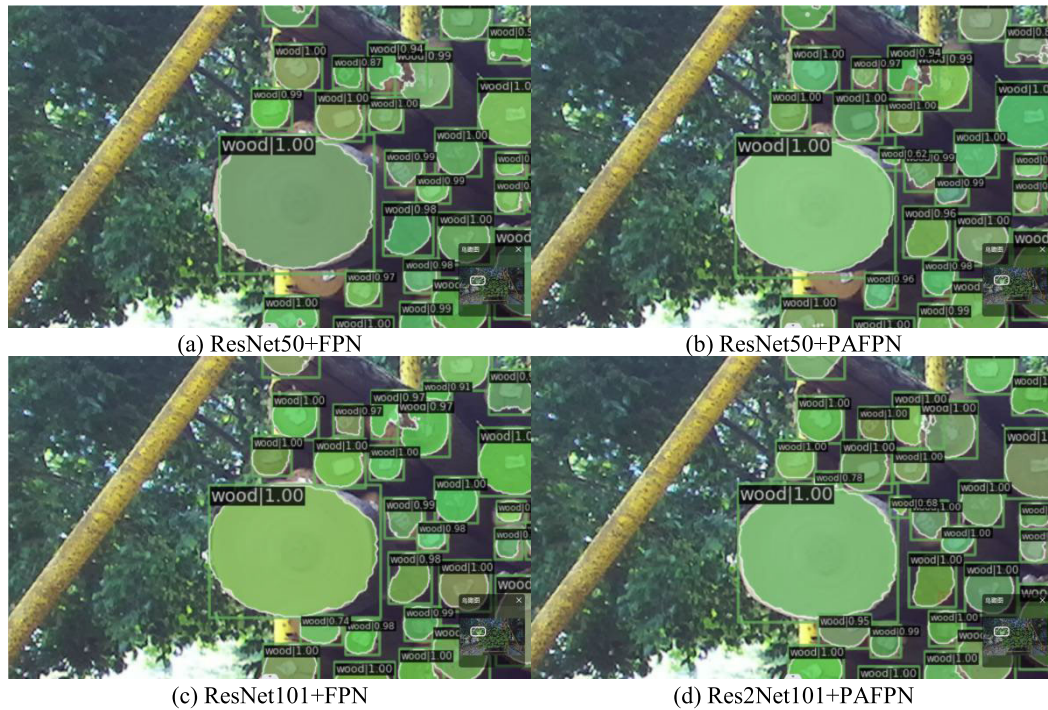
FIGURE 10. Backbone network improves model segmentation results of comparison experiments.

TABLE 6. The score of each group on each index in the test set.

Group	$mAP_{0.5:0.95}$ (%)	mAP_S (%)	mAP_M (%)	mAP_L (%)	$mIoU_{mask}$ (%)	Parameter	FPS
1	80.2	65.2	83.8	85.7	88.83	43.75M	3.9
2	80.5(+0.3)	64.9(-0.3)	84.0(+0.2)	86.1(+0.4)	89.09(+0.21)	47.29M(+8.09%)	3.7
3	80.9(+0.7)	65.4(+0.2)	84.6(+0.7)	88.8(+3.1)	89.02(+0.62)	62.74M(+43.41%)	3.2
4	81.8(+1.6)	**67.2(+2.0)**	85.0(+1.2)	**90.1(+4.4)**	**90.81(+2.60)**	66.94M(+53.01%)	2.9

the more layers of the neural network are deepened. However, increasing the layers sharply increase the number of model parameters from 43.75M to 62.74M (43.41%), and the model's inference speed decreased to 3.2FPS. Regarding group 4, logs' detection and segmentation performance is improved most obviously when the Res2Net structure and PAFPN structure are introduced simultaneously. Compared with the original Backbone structure of ResNet50+FPN, the segmentation accuracy of the improved model test set reached 81.8%, presenting a significant improvement of 1.6%. The segmentation accuracy of small, medium, and large woods is significantly improved, i.e., mAP_S, mAP_M, and mAP_L were improved by 2.0%, 1.2%, and 4.4%, respectively. Additionally, the $mIoU_{mask}$ scores also improved by 2.6%. The improved model combined with Res2Net and PAFPN further improves the wood detection and segmentation performance.

We uniformly adopted the classification confidence=0.6 threshold when visualizing the model segmentation results. The model segmentation results of the improved Backbone network and the competitors are illustrated in Figure 10,

revealing that in the inference results of the ResNet50+FPN model (Figure 10(a)), the recognition frame and mask could not completely cover the log end face area, and the obtained mask edge effect was jagged, presenting the worst performance. The inference result after replacing the FPN structure with the PAFPN structure is illustrated in Figure 10(b), which, compared with Figure 10(a), has a certain improvement. The mask and recognition box cover more comprehensive areas, but some missing segmentations remain. Figure 10(c) illustrates the reasoning performance of the experimental group that improved the model's feature extraction ability by increasing the network's layers. Compared with Figures 10(a) and 10(b), the missing segmentation phenomenon was improved, but part of the mask area exceeded the actual log end face. The fourth experiment was designed by combining the above three experimental groups, which involved a Backbone structure comprising Res2Net101 and PAFPN. The actual reasoning performance is depicted in Figure 10(d). At the same time, the edge of the mask is smooth. In conclusion, the ResNet101+PAFPN structure
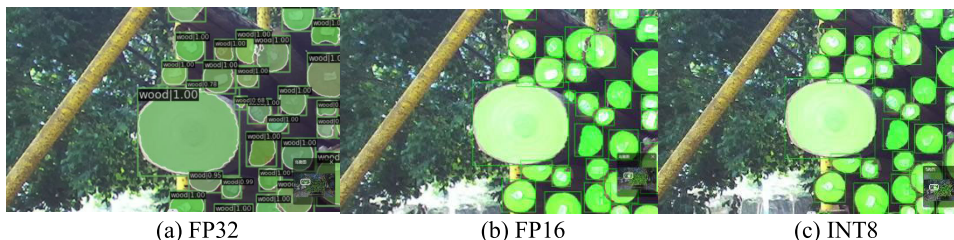
(a) FP32   (b) FP16   (c) INT8

FIGURE 11. Model inference results for different types of representation of weight.

TABLE 7. Comparison of the inference performance indexes of the weight parameters under various representation types.

	mAP0.5:0.95(%)	mAPS(%)	mAPM(%)	mAPL(%)	mIoUmask(%)
FP32	81.8	67.2	85.0	90.1	90.81
FP16	80.7(-1.1)	65.5(-1.7)	84.2(-0.8)	89.5(-0.6)	88.55(-2.26)
INT8	81.0(-0.8)	**64.5(-2.7)**	84.9(-0.1)	89.2(-0.9)	**88.27(-2.54)**

performs best on the segmentation mask compared to the combined model structure.

C. COMPARATIVE EXPERIMENT AND RESULT ANALYSIS ABOUT QUANTIFICATION

The general training model used 32-bit floating point numbers, considering that most models have a relatively strong anti-oise ability. Even if interference is added, the correct results are predicted, and thus the edge of the intelligent device decreases the precision and thus accelerates the calculations or reduces power consumption. The 32-bit floating point weight of each neural network layer was converted into an8-bit fixed point and min Max storage mode, significantly reducing the model's storage space and the memory access consumption during reasoning. However, the model's quantitative processing reduced its accuracy, affecting the final results. Thus, the following experiments were designed to explore the influence of quantification on the results. Considering the optimal training combination presented in Section III-B, i.e., the weight of the improved Mask R-CNN backbone with the Res2Net101+PAFPN combination as the quantization object. TensorRT as the quantization tool, and the test set of Section II-A as the test object, When the weights are FP32, FP16, and INT8, the performance of $mAP_{0.5:0.95}$, mAP_S, mAP_M, mAP_L, and $mIoU_{mask}$ on the test set was compared.

Specifically, Table 7 highlights tha after the weight of the Float32 representation was converted into a Float16 and INT8 representation, the model's final reasoning result was affected to some extent, and the value of each indicator decreased. INT8's mAP_S and $mIoU_{mask}$, reached 2.7% and 2.54%, respectively. Moreover, $mAP_{0.5:0.95}$, mAP_M, and mAP_L of INT8 were not particularly affected, which was less than 1%. Besides, the INT8's mAP_M, in particular, was 0.7% higher than FP16's 84.2% and just 0.1% lower than FP32's 85.0%. The process from Float32 to Int8 will narrow the numerical representation range, resulting in a loss of target segmentation accuracy of the model. This loss may cause

both the segmented mask area's narrowing and expansion. Through many experiments, quantization operation for the vast majority of logs makes the segment mask area of the end face smaller than the manually labeled area. An example is shown in Figure 11, where as the range of numerical representations decreases, the range of log face segmentation masks (green areas) decreases, especially at the edges.

Figure 11 presents the model's inference results, revealing that the reasoning results of FP16 and INT8 are consistent regarding the number of detections, and there is only a slight difference in the segmentation area of the mask compared to FP32. However, the difference can be controlled within a reasonable range. According to the analysis in Table 7 and Figure 11, the INT8 quantization operation on the model impacts the model's final inference result, but the impact can be controlled within a reasonable range, which is helpful for the deployment of the model in embedded edge devices.

NVIDIA's Jetson NX development board was taken as this paper's final target deployment platform. The following comparative experiment was set up to test the actual reasoning performance of the improved neural network model on the target platform. When testing the model's reasoning speed, the size of the model's input pictures was fixed to 1600*1216 pictures. The model was compared against NVIDIA's 3060 graphics card to evaluate the practical effect of deployment on embedded devices. Table 8 presents the corresponding results, revealing that the model's reasoning performance under different precision is very different under the same platform. As the weight representation's precision decreases, the model's storage space is significantly reduced. Specifically, the weight represented by FP32 was about 512MB, and by INT8 was only 96MB, which is only one-fifth of the FP32. The inference speed test of the model on the 3060 graphics card revealed that the inference speed of the weight represented by FP32 was only 2.9FPS while for INT8 was 22.878FPS, nearly 8 times faster. By comparing the reasoning performance of the INT8 models under different platforms, we found that the reasoning speed of the

TABLE 8. Model reasoning speed test under different hardware platform setups.

Platform	Power consumption	Weight type	Image size	Weight size	FPS
3060	170W	Fp32	1600x1216	512MB	2.9
		Fp16	1600x1216	145MB	15.304
		Int8	1600x1216	96MB	22.878
Jetson NX	20W	Fp32	1600x1216	512MB	0.3
		Fp16	1600x1216	145MB	1.679
		Int8	1600x1216	96MB	2.523

3060 graphics card with a high computing power reached 22.878FPS while the Jetson NX platform with low computing power was 2.523FPS, far less than the 3060. However, when testing the power consumption of the two platforms, we found that the Jetson NX's power consumption was only 20W, much smaller than the 3060's power consumption. We quantified the speed increase on the Jetson NX development board, from 0.3FPS on FP32 to 2.523FPS on INT8, a speed increase of more than eight times.

D. ANALYZING THE CALCULATION RESULTS OF VEHICULAR LOG VOLUME

The INT8 quantification network model of the improved Mask R-CNN of the Res2Net101+PAFPN structure, which presented the best performance inSection B, was selected to complete the calculation of the volume of the whole vehicle logs. Thus, we considered cars of logs and designed the comparative experiment presented in Table 9. Specifically, two experienced workers were asked to examine 5 carloads of logs separately. After calculating the volume, the average value was used as the ground truth. Then, the model without quantization acceleration and INT8 quantization were used to complete the recognition of the whole vehicle log to obtain the mask map of the log end face. The ellipse fitting method described in Section II-C-I and the diameter estimation method of Section II-C-II were used to obtain the inspection diameter of the log. Then the volume was calculated using the same method as the manual inspection ruler. The volume obtained by using the weight expressed by FP32 is reported in Table 9. Compared with the manual inspection result, the average error of the log volume of 10 cars is 1.5206%, and the maximum error is 3.5444%. Compared with the manual measurement results, the average error of the network model volume prediction results represented by INT8 without any compensation was 2.5450%. Compared with the results of FP32, the average error was 1.0244% higher, and the maximum error reached 4.4424%. The results proved that quantifying neural networks affects the final detection results. By observing the volume calculation results of each vehicle and the mask segmentation effect illustrated in Figure 11, we find that:the volume calculation result of INT8 is always less than the FP32, with Figure 11 illustrating the reason. Quantizing the neural network further

reduces the segmentation area of the mask, decreasing the overall volume. To reduce the accuracy of the quantitative comparison, we conducted another set of experiments that divided the log mask placed not immediately to the fitting operation but to the masked figure for inflation convolution operation. Thus, expanding the convolution operation can increase the mask of the segmented regions, thus reducing the mask precision compensation of the quantitative losses. The volume results obtained after expansion convolution are reported in Table 9. The average error of the vehicle volume was 0.646%, and the maximum error was only 1.6833%, which decreased by 1.8982% and 2.7591%, respectively, compared with the calculation results before compensation, proving that the expansion convolution can compensate for the quantization loss.

IV. DISCUSSION

From the test data in Table 6, the segmentation detection performance of the improved Mask R-CNN instance segmentation network is improved to a certain extent. Especially for small and large woods, mAP increased by 2% and 4.4%, respectively. In the test set, the mask segmentation accuracy improved by 2.6% to 90.81%, highlighting that the effect of the improved model is not only reflected through the indexes. Figure 10 illustrates the reasoning results of the improved model in more intuitive details, revealing that even a partially occluded model can detect wood. Compared with the original Mask R-CNN, the segmentation accuracy of a mask is also improved to a certain extent, and the obtained mask is more consistent with the log face contour. Here, although the model's detection performance is improved, the inference speed reduces as the model size increases, affecting the algorithm's execution efficiency.

Moreover, to meet the needs of detection anytime and anywhere, this paper explores the actual operation effect of the model on embedded devices based on a Jetson NX development board. Table 8 compares the performance differences between Jetson NX and an NVIDIA 3060 graphics card, revealing that for reasoning speed, the 3060 graphics card with high computational power outperforms the embedded devices that affor a low computational power requiremen. However, embedded devices have their advantages, as their power consumption is very low, and they

TABLE 9. Actual volume measurement results.

Group	Artificial Result(m³)	FP32 Result(m³)	Error (%)	None Processing result		Dilated Convolution Result	
				INT8 Result(m³)	Error (%)	INT8 Result(m³)	Error (%)
1	6.4326	6.4301	0.0389	6.4236	0.1399	6.4091	0.3653
2	7.1090	7.0714	0.5289	6.8577	3.5350	7.1572	0.6773
3	6.8770	6.8135	0.9234	6.7745	1.4905	6.9035	0.3853
4	5.8599	5.6522	3.5444	5.6497	3.5871	5.9325	1.2389
5	6.2091	6.0707	2.2282	5.9428	4.2881	6.2346	0.4107
6	5.4583	5.4511	0.1319	5.4389	0.3554	5.4833	0.4580
7	5.7819	5.6837	1.6984	5.6575	2.1446	5.7548	0.4687
8	6.4965	6.2988	3.0432	6.2079	4.4424	6.5325	0.5541
9	6.9981	6.9543	0.6259	6.8938	1.4904	7.1159	1.6833
10	6.3652	6.2097	2.4430	6.1121	3.9763	6.3796	0.2262
	Average Error(%)		1.5206		2.5450		0.6468

are suitable for mobile development. Hence, to improve the model's reasoning speed on embedded devices, this paper uses the TensorRT tool to quantify the model, and the model's reasoning speed increased by 8 times after quantization. Although quantization improves the model's reasoning speed, it inevitably affects its detection and segmentation performance.

From the performance of the actual reasoning results in Figure 11, the quantization makes the mask region segmented by the model smaller. Therefore, during image post-processing, this paper adopts the expansion convolution operation to compensate for the loss of mask segmentation accuracy caused by quantization and experiments on the data of 5 cars of logs. Table 9 highlights that compared with the results without processing, the volume data obtained after expansion convolution compensation has a lower error, and the average error of the final volume calculation result is controlled within 1%.

The diameter data of the log face are obtained by ellipse fitting based on the masking diagram of the model inference results. Although certain results have been obtained, the data obtained in some cases are inaccurate, e.g., under partial occlusion, often seen in the detection of vehicle logs. Therefore, in practical applications, the proposed algorithm must conduct a certain specification of log loading and expose all logs as far as possible. The detection of partially occluded logs is also our next step.

V. CONCLUSION

In the process of forestry production, the measuring scale of logs is the key to the efficient utilization of wood resources. The measuring scale accuracy of the logs determines the value of the wood. To solve the problems of low efficiency, high cost, and high-risk coefficient of manual inspection ruler, this paper designs and implements an intelligent volume calculation method of vehicular log based on an improved Mask R-CNN instance segmentation model. Moreover, we successfully deployed our algorithm on an embedded platform, contributing valuable experience for the intelligent development of forestry log inspection ruler.

Specifically, this paper uses Mask R-CNN as the basic model to train the collected vehicle log dataset and evaluate it on the test set. The backbone network of the original algorithm is improved further to improve the model's detection and segmentation performance. By introducing the Res2Net and PAFPN structures in the Mask R-CNN, the network's feature map extraction ability for logs of various sizes is enhanced, and the comprehensive performance of the network is further improved. The performance comparison experiment on the test set verified the improved detection accuracy and masked segmentation quality score of the introduced module compared with the original model. Based on the mask map, the LSM ellipse fitting method is used to obtain the diameter data of logs in the image. Aiming at the stratification problem encountered in practical engineering applications, an adaptive stratified log diameter estimation method based on K-means and depth information fusion is proposed, which is successfully applied to the actual measuring operation. In order to deploy the method proposed in this paper in embedded devices, the neural network model is compressed based on a quantization method, and the dilated convolution operation is proposed to compensate for the loss of a mask segmentation region.

Extensive evaluation of the INT8 quantification network model utilizing the improved Mask R-CNN based on the Res2Net101+PAFPN structure and applied on the Jetson NX embedded development platform demonstrates our method's capabilities attaining an average and a maximum vehicle volume error of 0.646% and 1.6833%, respectively.

The embedded vehicle-mounted log intelligent measuring method designed and implemented in this paper meets the actual measuring application needs and reduces labor costs. It should be noted that the suggested method can adapt to different log detection places with a few changes and has a wide application prospect. The development of intellectualized and informationalized level of the log checking ruler will be helpful to improve the efficiency of log production and utilization rate of wood. Therefore, in the future, we will develop a simple human-computer interaction

interface and log information management platform based on this study to provide a solution for intelligent log management.

REFERENCES

[1] K. Fang, C. Li, Y. Tang, J. He, and J. Song, "China's pathways to peak carbon emissions: New insights from various industrial sectors," *Appl. Energy*, vol. 306, Jan. 2022, Art. no. 118039, doi: 10.1016/j.apenergy.2021.118039.

[2] Z. Zheng, "Re-calculation of responsibility distribution and spatiotemporal patterns of global production carbon emissions from the perspective of global value chain," *Sci. Total Environ.*, vol. 773, Jun. 2021, Art. no. 145065, doi: 10.1016/j.scitotenv.2021.145065.

[3] M. Tavoni, B. Sohngen, and V. Bosetti, "Forestry and the carbon market response to stabilize climate," *Energy Policy*, vol. 35, no. 11, pp. 5346–5353, Nov. 2007, doi: 10.1016/j.enpol.2006.01.036.

[4] R. Ruotsalainen, T. Pukkala, A. Kangas, M. Myllymäki, and P. Packalen, "Economic losses in carbon forestry due to errors in inventory data," *Can. J. Forest Res.*, vol. 51, no. 4, pp. 501–512, Apr. 2021, doi: 10.1139/cjfr-2020-0251.

[5] A. Duka, M. Sertić, T. Pentek, I. Papa, D. Janeš, and T. Poršinsky, "Round wood waste and losses—Is rationalisation in scaling possible?" *Croatian J. Forest Eng., J. Theory Appl. Forestry Eng.*, vol. 41, no. 2, pp. 1–12, Jun. 2020, doi: 10.5552/crojfe.2020.770.

[6] T. Gergel, T. Bucha, R. Gracovský, M. Chamula, M. Gejdoš, and P. Veverka, "Computed tomography as a tool for quantification and classification of roundwood—Case study," *Forests*, vol. 13, no. 7, p. 1042, Jul. 2022, doi: 10.3390/f13071042.

[7] S.-W. Hwang, T. Lee, H. Kim, H. Chung, J. G. Choi, and H. Yeo, "Classification of wood knots using artificial neural networks with texture and local feature-based image descriptors," *Holzforschung*, vol. 76, no. 1, pp. 1–13, Jan. 2022, doi: 10.1515/hf-2021-0051.

[8] S. Yella and M. Dougherty, "Automatically detecting the number of logs on a timber truck," *J. Intell. Syst.*, vol. 22, no. 4, pp. 417–435, Dec. 2013, doi: 10.1515/jisys-2013-0026.

[9] J. Li, H. Xu, Y. Yu, H. Chen, W. Yi, and H. Wang, "Intelligent analysis technology of bamboo structure. Part I: The variability of vascular bundles and fiber sheath area," *Ind. Crops Products*, vol. 174, Dec. 2021, Art. no. 114163, doi: 10.1016/j.indcrop.2021.114163.

[10] H. Zhao, C. Wang, R. Guo, X. Rong, J. Guo, Q. Yang, L. Yang, Y. Zhao, and Y. Li, "Autonomous live working robot navigation with real-time detection and motion planning system on distribution line," *High Voltage*, vol. 7, no. 6, pp. 1204–1216, Dec. 2022, doi: 10.1049/hve2.12221.

[11] Z. A. Abbood, I. Khaleel, and K. Aggarwal, "Challenges and future directions for intrusion detection systems based on AutoML," *Mesopotamian J. Cybersecur.*, vol. 2021, pp. 16–21, Jan. 2021, doi: 10.58496/MJCS/2021/004.

[12] R. T. Rasheed, Y. Niu, and S. N. Abdl, "Harmony search for security enhancement," *Mesopotamian J. Cybersecur.*, vol. 2021, pp. 5–8, Jan. 2021.

[13] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[14] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 2778–2788.

[15] 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, Jun. 2017, pp. 1–15.

[16] B. Galsgaard, D. H. Lundtoft, I. Nikolov, K. Nasrollahi, and T. B. Moeslund, "Circular Hough transform and local circularity measure for weight estimation of a graph-cut based wood stack measurement," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 686–693, doi: 10.1109/WACV.2015.97.

[17] F. Budiman, R. Mardiyanto, and R. Rachmat, "A handy and accurate device to measure smallest diameter of log to reduce measurement errors," in *Proc. Int. Seminar Intell. Technol. Appl. (ISITIA)*, Jul. 2016, pp. 423–428, doi: 10.1109/ISITIA.2016.7828697.

[18] A. V. Kruglov, "The algorithm of the roundwood volume measurement via photogrammetry," in *Proc. Int. Conf. Digit. Image Comput., Techn. Appl. (DICTA)*, Nov. 2016, pp. 1–5, doi: 10.1109/DICTA.2016.7797088.

[19] A. Kruglov and Y. Chiryshev, "The image analysis algorithm for the log pile photogrammetry measurement," *WSEAS Trans. Signal Process.*, vol. 13, pp. 135–145, Apr. 2017.

[20] A. Kruglov and E. Shishko, "Log pile measurement through 3D modeling," in *Proc. 40th Int. Conf. Telecommun. Signal Process. (TSP)*, Jul. 2017, pp. 263–266, doi: 10.1109/TSP.2017.8075983.

[21] Y. V. Chiryshev, A. V. Kruglov, and A. S. Atamanova, "Automatic detection of round timber in digital images using random decision forests algorithm," in *Proc. Int. Conf. Control Comput. Vis.*, 2018, pp. 39–44, doi: 10.1145/3232651.3232667.

[22] M. Acuna and A. Sosa, "Automated, volumetric measurements of truckloads through multi-view photogrammetry and 3D reconstruction software," *Croatian J. Forest Eng., J. Theory Appl. Forestry Eng.*, vol. 40, no. 1, pp. 151–162, 2019.

[23] E. Gutzeit and J. Voskamp, "Automatic segmentation of wood logs by combining detection and segmentation," in *Proc. Int. Symp. Vis. Comput.* Cham, Switzerland: Springer, 2012, pp. 252–261, doi: 10.1007/978-3-642-33179-4_25.

[24] C. Herbon, K. Tönnies, and B. Stock, "Detection and segmentation of clustered objects by using iterative classification, segmentation, and Gaussian mixture models and application to wood log detection," in *Proc. German Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2014, pp. 354–364, doi: 10.1007/978-3-319-11752-2_28.

[25] H. Tang, K. Wang, J. Gu, X. Li, and W. Jian, "Application of SSD framework model in detection of logs end," *J. Phys., Conf. Ser.*, vol. 1486, no. 7, 2020, Art. no. 072051, doi: 10.1088/1742-6596/1486/7/072051.

[26] N. Samdangdech and S. Phiphobmongkol, "Log-end cut-area detection in images taken from rear end of eucalyptus timber trucks," in *Proc. 15th Int. Joint Conf. Comput. Sci. Softw. Eng. (JCSSE)*, Jul. 2018, pp. 1–6, doi: 10.1109/JCSSE.2018.8457388.

[27] P. Yang, J. Zheng, Z. Feng, Z. Ding, S. Li, Q. Huang, and L. Kong, "Research on detection and segmentation methods of dense-stacked logs using Mask R-CNN," *J. Forestry Eng.*, vol. 7, no. 2, pp. 135–142, 2022, doi: 10.13360/j.issn.2096-1359.202106001.

[28] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. ICCV*, 2017, pp. 2961–2969, doi: 10.1109/ICCV.2017.322.

[29] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, Jun. 2015, pp. 1–13.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[31] X. Li, N. Li, C. Weng, X. Liu, D. Su, D. Yu, and H. Meng, "Replay and synthetic speech detection with Res2Net architecture," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 6354–6358, doi: 10.1109/ICASSP39728.2021.9413828.

[32] N. E. Ocer, G. Kaplan, F. Erdem, D. K. Matci, and U. Avdan, "Tree extraction from multi-scale UAV images using mask R-CNN with FPN," *Remote Sens. Lett.*, vol. 11, no. 9, pp. 847–856, Sep. 2020, doi: 10.1080/2150704X.2020.1784491.

[33] S. G. S. Guo, Z. W. S. Guo, Y. L. Z. Wang, X. L. Y. Lou, and H. L. X. Li, "Detection method of photovoltaic panel defect based on improved mask R-CNN," *J. Internet Technol.*, vol. 23, no. 2, pp. 397–406, Mar. 2022, doi: 10.53106/160792642022032302018.

[34] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768, doi: 10.1109/CVPR.2018.00913.

[35] S. Lou, F. Liu, and H. Wang, "Error analysis of cycloid gear based on ellipse fitting based on least square approach," *J. Phys., Conf. Ser.*, vol. 2095, no. 1, 2021, Art. no. 012067, doi: 10.1088/1742-6596/2095/1/012067.

[36] K. Chen et al., "MMDetection: Open MMLab detection toolbox and benchmark," 2019, *arXiv:1906.07155*.

[37] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, "A survey on modern trainable activation functions," *Neural Netw.*, vol. 138, pp. 14–32, Jun. 2021, doi: 10.1016/j.neunet.2021.01.026.

[38] Z. Lou, W. Zhu, and W. B. Wu, "Beyond sub-Gaussian noises: Sharp concentration analysis for stochastic gradient descent," *J. Mach. Learn. Res.*, vol. 23, pp. 1–22, Jun. 2022.

JISHI ZHENG received the Ph.D. degree from Central South University, China, in 2015. From January to July 2019, he was a Visiting Scholar at the Robotics Laboratory, Computer and Electronic Engineering Department, University of Essex, U.K. He is with the Fujian University of Technology. He is currently the Head of the Internet of Things major, the Director of the Teaching and Research Department of Traffic Information and Control, and the Executive Director of the Fujian Society of Aeronautics. He has presided over and participated in more than ten provincial and municipal scientific research projects. His main research interests include application of artificial intelligence in the industry and research of UAV flight control algorithm.

SHAOYI LI was born in Quanzhou, Fujian, China, in 1997. He is currently pursuing the M.A.Eng. degree with the School of Transportation, Fujian University of Technology, China. His research interests include image processing and machine learning.

SHIWEN ZHANG is currently pursuing the M.A.Eng. degree with the School of Transportation, Fujian University of Technology, China. His research interest includes application of artificial intelligence in forestry.

LINGHUA KONG received the Ph.D. degree in mechanical engineering from McGill University, Canada, in 2004. He has been with the Fujian University of Technology, since 2015. He has designed and developed a variety of new products and equipment and received 15 patents; As the lead author, he has published 20 influential papers included in SCI/EI.

DING ZHIGANG received the master's degree from Jilin University, China, in 2007. He is with the Fujian University of Technology. He is a Master's Supervisor and currently the Head of the Vehicle Engineering Laboratory. He has participated in National 863 Program and more than ten provincial and municipal scientific research projects. His main research interest includes application of artificial intelligence in the vehicle industry.

• • •