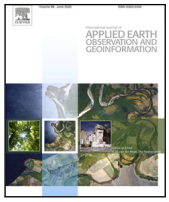
International Journal of Applied Earth Observation and Geoinformation 131 (2024) 103923

Contents lists available at ScienceDirect

International Journal of Applied Earth Observation and Geoinformation

journal homepage: www.elsevier.com/locate/jag



VrsNet - density map prediction network for individual tree detection and counting from UAV images

Taige Luo [a,1], Wei Gao [a,1], Alexei Belotserkovsky [c], Alexander Nedzved [d], Weijie Deng [e], Qiaolin Ye [a,f,g,h], Liyong Fu [b], Qiao Chen [b], Wenjun Ma [b], Sheng Xu [a,*]

[a] College of Information Science and Technology & Artificial Intelligence, Nanjing Forestry University, Nanjing, 210037, Jiangsu, China
[b] Institute of Forest Resource Information Techniques Chinese Academy of Forestry, Beijing, 100091, China
[c] United Institute of Informatics Problems, National Academy of Sciences of Belarus, Minsk, 220012, Belarus
[d] Belarusian State University, Minsk, 220030, Belarus
[e] College of Economics and Management, Nanjing Forestry University, Nanjing, 210037, Jiangsu, China
[f] State Key Laboratory of Tree Genetics and Breeding, Nanjing Forestry University, Nanjing, 210037, Jiangsu, China
[g] Co-Innovation Center for Sustainable Forestry in Southern China, Key Laboratory of Tree Genetics and Biotechnology of Educational Department of China, Nanjing Forestry University, Nanjing, 210037, Jiangsu, China
[h] Key Laboratory of Tree Genetics and Silvicultural Sciences of Jiangsu Province, Nanjing Forestry University, Nanjing, 210037, Jiangsu, China

ARTICLE INFO

Keywords:
Vegetation mapping
Remote sensing
Semi-supervised learning
Object detection
Tree segmentation
Tree counting
Density map

ABSTRACT

Individual tree detection and counting in unmanned aerial vehicle (UAV) imagery constitute a vital and practical research field. Vegetation remote sensing captures large-scale trees characterized by complex textures, significant growth variations, and high species similarity within the vegetation, which presents significant challenges for annotation and detection. Existing methods based on bounding boxes have struggled to convey semantics information about tree crowns. This paper proposes a novel deep learning network called VrsNet based on the density map information. The proposed work pioneers the segmentation and counting application by utilizing the semantic information of Gaussian contour. Besides, we sample and create the UAV vegetation remote sensing density dataset TreeFsc for experiments. In quantitative comparison across multiple datasets, the proposed method demonstrates high performance, with a 3.45 increase in MAE and a 4.75 increase in RMSE. Experiments demonstrate superior cross-region, cross-scale, and cross-species target detection capabilities of the proposed approach compared with the existing object detection methods. Our code and dataset are available at: https://github.com/luotiger123/VrsNet/tree/main/VrsNet.

1. Introduction

In recent years, the vegetation remote sensing images have expanded into various fields, including grouping (Liu et al., 2012), segmentation (Cheng et al., 2023; Zhao et al., 2024; Nasiri et al., 2023), individual tree research (Jiang et al., 2023; Hui et al., 2022; Zheng et al., 2023), and visual analysis (Zang et al., 2024). In individual tree detection, high similarity and inter-species heterogeneity present challenges for existing neural network models to extract vegetation textural features. Traditional bounding-box-based detection methods require tedious annotation of individual tree regions, resulting in significant errors and a substantial workload. Conversely, current density-based detection methods struggle to capture the contour information of vegetation. Therefore, this study proposes a novel semi-supervised deep learning model for estimating the density distribution.

Vegetation sampled by drones presents three challenges compared with indoor images (Tolan et al., 2024): Firstly, drones are sensitive to changes in spatial scales. This sensitivity leads to more diverse spatial characteristics. Secondly, vegetation exhibits high inter-species heterogeneity. This demands that the model extracts various textures and crown diameters. Lastly, vegetation grows in various environments, which requires the model to possess cross-regional recognition capabilities. To address the above issues, we introduce the novel Mapping and Correlation module (MAC) for correlated feature extraction. Additionally, we propose a Multi-layer Coordinate Attention (MLCA) as an intrinsic attention mechanism. This mechanism enhances our model's capability for cross-region recognition. Finally, we propose an Adaption (Ap) module for lightweight parameter fine-tuning during testing.

* Corresponding author.
E-mail addresses: chenq@ifrit.ac.cn (Q. Chen), xusheng@njfu.edu.cn (S. Xu).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.jag.2024.103923
Received 24 February 2024; Received in revised form 15 April 2024; Accepted 17 May 2024
Available online 2 June 2024

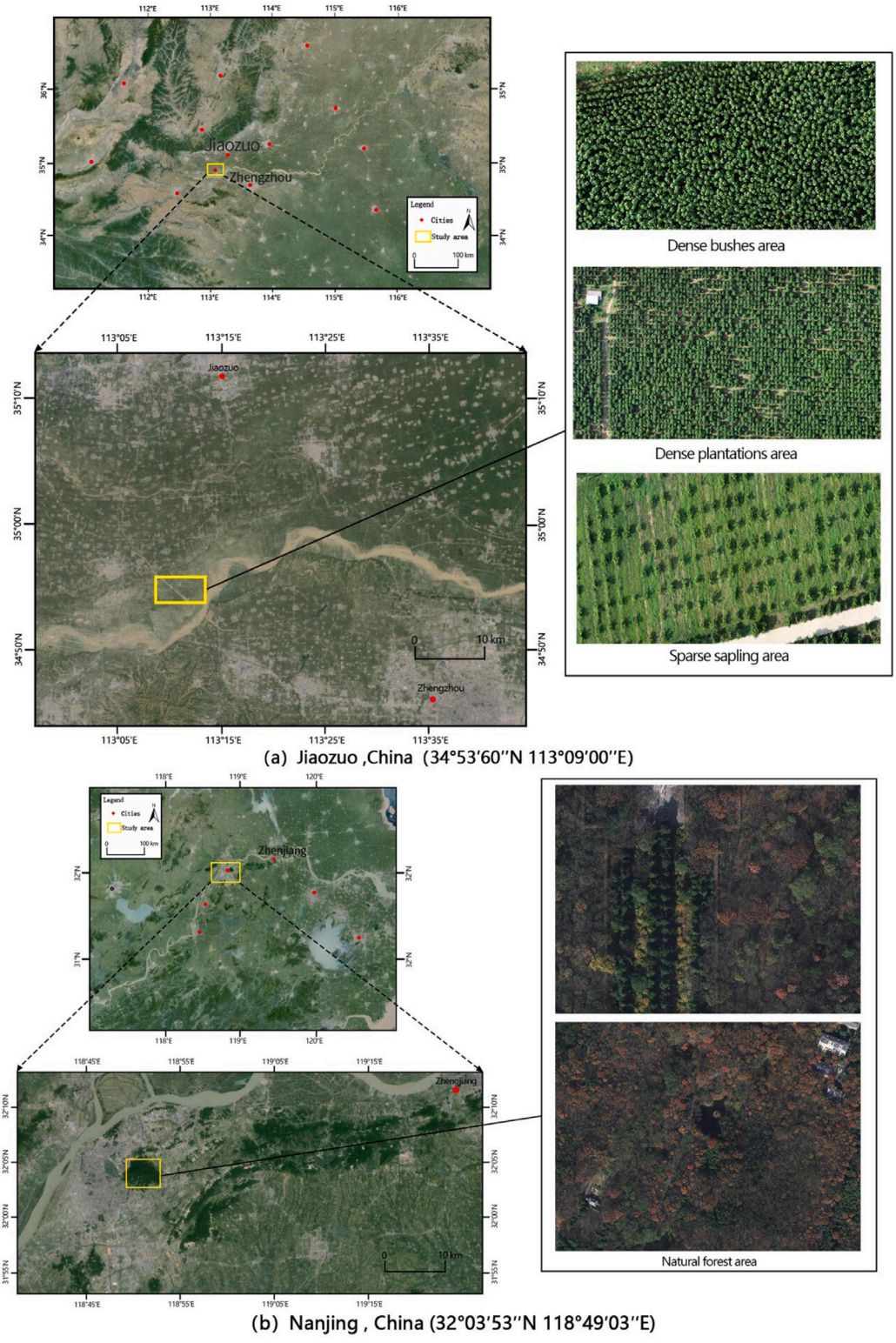


Fig. 1. Study areas.

These architectures enable the network to achieve better predictions of density distribution maps.

Our experiments regions have distinct geomorphic features as shown in Fig. 1. Firstly, we conduct experiments in artificial planting forests located in Jiaozuo, China ($34°53′60″N, 113°09′00″E$). The area boasts diverse topography, abundant mountainous resources, and a humid, warm climate. The area is populated with catalpa trees in different stages of growth. The challenging terrain hampers the utilization of vegetation resources. Secondly, we select Xuanwu Lake Natural Forest Park in Nanjing, China, as our study site ($32°03′53″N, 118°49′03″E$). The area is populated with many common deciduous trees such as cedar, cypress, camphor, and chestnut. The diverse vegetation not only holds significant research value but also presents challenges to the model's compatibility. Our dataset TreeFsc is constructed based on

two research areas. Fig. 1. fully illustrates the diversity of our dataset, which is the first density dataset designed for individual-tree extraction research. Our model does not rely on sampling distance and ensures a spatial resolution of approximately 30 centimeters.

Our contributions can be summarized as follows:

- we pioneer a two-step individual tree segmentation and counting method, applying density map prediction to individual tree detection research.
- we propose a novel semi-supervised network that focuses on the extraction of correlated features, which is more suitable for the detection and complex vegetation research.
- we open-source a new drone remote sensing vegetation dataset TreeFsc, and provide density annotations, facilitating research migration. Experiments demonstrate that our model predicts more accurate density distribution maps for various objects, facilitating object detection and counting studies compared with other state-of-the-art methods.

2. Related work

2.1. Bounding-box-based object detection methods

Traditional bounding-box-based object detection methods integrate convolutional neural networks into detection. For instance, R-CNN (Girshick et al., 2014) summarizes the detection in 3 steps: first, generating candidate regions using Selective Search; then, extracting convolutional features for each region; finally, performing object detection using classifiers and regressors. However, processing a large number of candidate regions resulted in slow network speed. Later, Faster R-CNN, as proposed by Ren et al. (2015), addressed this limitation by introducing the Region Proposal Network (RPN), which generates candidate regions, resulting in an accurate and efficient end-to-end network. Subsequently, He et al. (2017) enhanced Faster R-CNN by introducing a segmentation head, resulting in Mask R-CNN. This endows Mask R-CNN with the capability of instance segmentation. Then, the YOLO series proposed by Redmon et al. (2016) introduces a one-stage detection approach, transforming the object detection task into a regression problem. However, we find that methods based on bounding boxes are not suitable for vegetation detection. Vegetation displays high scale variations, intensive scenes and complex texture features, which poses challenges for bounding box models.

2.2. Density-based object detection methods

Recently, scholars have increasingly focused on the localization and counting of density distribution maps, achieving notable progress in areas such as crowds (Liu et al., 2019; Boominathan et al., 2016; Li et al., 2018; Zhang et al., 2016; Ranasinghe et al., 2023), vehicles (Mundhenk et al., 2016; Guo et al., 2022), cells (Jayarao et al., 2004), pests (Zhang et al., 2024), and multi-class counting (Ranjan et al., 2021; Xu et al., 2021a).

The density map represents the probability distribution of vegetation presence at each pixel. The values range between 0 and 1, with those closer to 1 indicating a higher likelihood of vegetation presence. The density generation process is inspired by the method proposed by Zhang et al. (2016). This approach ensures a uniform gradient nature of the density distribution image. This characteristic allows our density map to provide improved estimations of individual trees.

In the early stages of CNN-based density counting networks, a common design choice is to utilize a single-branch structure. For example, Crowd CNN (Wang et al., 2015), focused on generating crowd density maps to achieve counting results. Due to the presence of multi-scale variations in real-world images, Boominathan et al. (2016) introduces the multi-branch structure CrowdNet, which combines deep and shallow sub-network structures to perceive the uneven scale variations of crowd density. Inspired by multi-branch neural networks, Zhang et al. (2016) proposed the multi-column convolutional neural network MCNN, which uses columns with different-sized convolutional kernels to extract feature information. However, due to the excessive parameters in the multi-column convolution, Khoo and Ying (2019) introduces the SwitchNet, which predicts the density level of the image itself through a classifier for density regression.

Subsequently, researchers are no longer satisfied with the high redundancy of multi-column network architectures. For example, Li et al. (2018) introduces the dilated convolution network CsrNet. The application of dilated convolution maintains resolution while expanding the advantages of the receptive field, thereby preserving more image details. Liu et al. (2019) proposes the Context Aware Network CAN, which reflects the scale information of objects. Then, Ranjan et al. (2021) proposes a few-shot density counting method FamNet. Although this structure takes local features into account, it does not involve attentional filtering of regions. The reason is that those methods focus on fusing global features with local features without considering their implicit connections.

2.3. Individual-tree segmentation methods

Individual-tree crown segmentation is widely used in combination with point cloud semantic segmentation (Xu et al., 2018, 2023, 2021b). Its core workflow involves associating pixel values with semantic information in the point cloud. Forest remote sensing monitoring requires geometric structural information based on trees, which cannot be directly obtained from a single frame or unit. With the development of deep learning, more and more scholars are adopting deep neural networks to process individual-tree crown remote sensing information and achieve high-quality results (Pu et al., 2022). However, scholars lack precedents for applying density maps to vegetation detection, and the existing vegetation segmentation methods require precise pixel-level annotations, resulting in huge workloads, which will be addressed in our work.

3. The proposed method

Our workflow is depicted in Fig. 2. Next, we introduce the density map generation algorithm in detail. We present the general framework of the VrsNet model from a holistic perspective as shown in Fig. 3. Subsequently, we focus on introducing the key modules: the Mapping and Correlation module (MAC), the Multi-layer Coordinate Attention (MLCA), and the Adaption module (Ap). Additionally, we elaborate on the employed loss functions in this approach and we analyze the algorithm's parameters and complexity comprehensively.

3.1. Our work

Our work can be summarized into two processes: network prediction and downstream applications as shown in Fig. 2.

Firstly, we obtain the individual tree density detection map through the proposed VrsNet. The network process is illustrated within the blue dashed box. The original input image is predicted through various network modules to obtain a preliminary density map denoted as Pred. By comparing with the Ground Truth in the Ap module, the regression module parameters are adjusted, yielding a more accurate density map. We conduct two different downstream tasks based on the obtained density map as shown in the red dashed box. To fully leverage the Gaussian semantic information implied in the vegetation density map. The first branch is tree crown segmentation application: by applying hierarchical filtering to the density map, a vegetation density contour map can be derived. Then, by controlling threshold parameters, mask images of different intensities can be generated and can be used for vegetation contour segmentation applications. The second branch is the large-scale high-density spatial vegetation counting application:

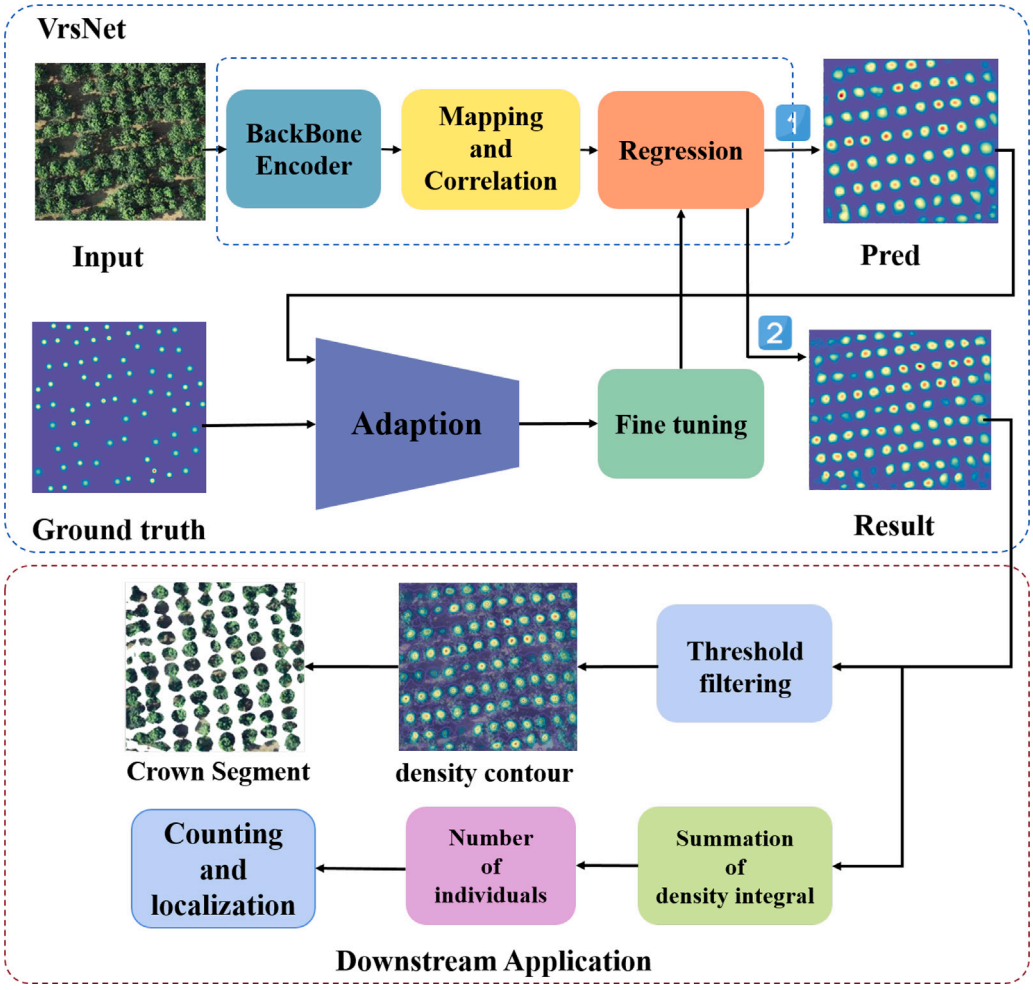


Fig. 2. The proposed work. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

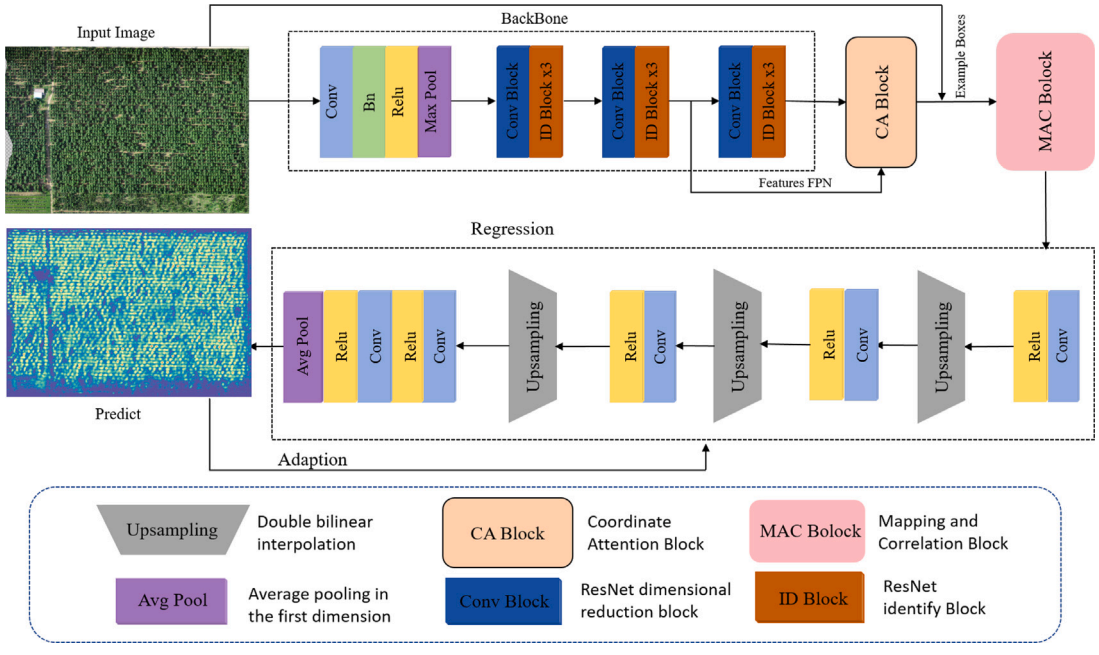


Fig. 3. The network architecture.

the probability density map contains information about vegetation distribution. By integrating and summing, the number of vegetation trees can be estimated. These achievements will be further discussed in the following sections.

3.2. Density map generation algorithm

According to Zhang et al. (2016), we initially generate a full-zero matrix of the same size as the original image and mark the positions x_i corresponding to the centers of targets. This process can be described as follows:

$$\mathcal{H}(x) = \sum_{i=1}^{N} \delta(x - x_i), \tag{1}$$

where x_i denotes the coordinate of targets, $\delta(x - x_i)$ represents the impulse function to mark the center of the target as 1 in the matrix, while the rest is marked as 0.

Subsequently, we obtain distinct regions for each x_i. Choose an appropriately sized Gaussian kernel enables its range to accurately represent the object's contour size. The formula is given by:

$$F(x) = \sum_{i=1}^{n} \delta(x - x_i) \times G_{\sigma_i}(x), \ with \ \sigma_i = \beta \bar{d}^i, \tag{2}$$

$$\bar{d}^i = \frac{1}{m} \sum_{j=1}^{m} d_j^i, \tag{3}$$

where $G_{\sigma_i}(x)$ represents the size of Gaussian kernel. They calculate the average distance $\bar{d}^i$ for m points to determine the Gaussian kernel size, ensuring the accuracy of the density map. β is set to 0.3 as a control constant.

The density distribution map reveals implicit semantic information concerning vegetation contours. We propose a novel approach for individual tree segmentation based on the uniform gradient characteristics of the density distribution maps.

3.3. Overview

In our method, we have completed the prediction process from original vegetation to density maps as shown in Fig. 3. Our approach employs a semi-supervised learning method, where the network inputs the original remote sensing image $x \in \mathbb{U}^{H \times W \times 3}$, along with a small number of annotated example box coordinates B. The output of this network is the predicted density map $Y \in \mathbb{U}^{H \times W}$. Firstly, we obtain multi-layer features M_i through the ResNet101FPN (He et al., 2016) backbone network. Subsequently, by passing these multi-layer features through the Multi-layer Coordinate Attention (MLCA), we obtain a global feature layer F_i as shown in Fig. 6. The multi-layer features are then sequentially input into the MAC module as shown in Fig. 4. Initially, a coordinate mapping and upsampling layer pool the features to a uniform size as shown in Fig. 5. Different factors are applied to scale the example features in order to obtain local features E_i. These local features E_i are then convoluted with F_i through self-convolutional operations. This process results in correlated feature layers C_i. The final correlated feature map C is obtained by fusing these relevant feature layers C_i through max-pooling. Fused correlated C is then input into our regression layer. We further refine the model parameters through the Adaption module(Ap) to obtain the final output density map Y. The detailed process is as follows:

$$M_i = \mathcal{F}(\mathfrak{B}_{101}(x), i), \tag{4}$$

$$F_i = MLCA(M_i), \tag{5}$$

where $\mathfrak{B}_{101}(x)$ denotes the set of Multi-features layers obtained by processing the input image x through the ResNet101. $\mathcal{F}$ refers to the Feature Pyramid Network (FPN) operation. i refers to the ith feature layer of Multi-features layers. $MLCA$ stands for performing Coordinate Attention learning on each layer of feature maps obtained through FPN operations.

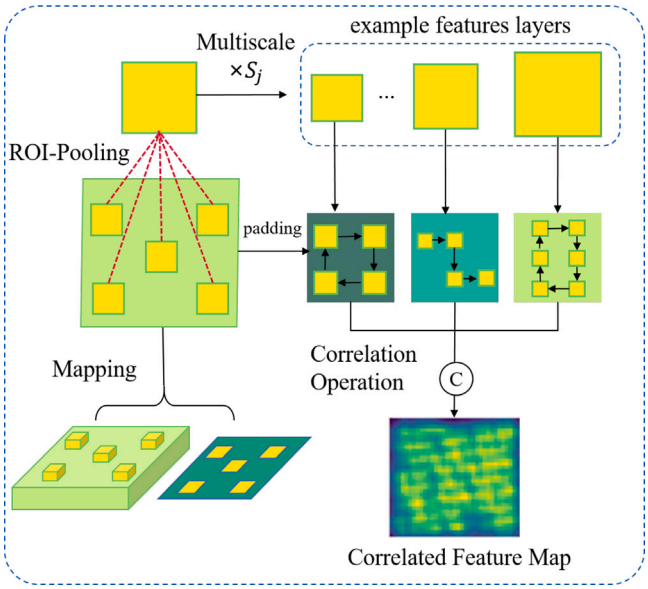


Fig. 4. Mapping and correlation block.

Fig. 5. Flowchart for obtaining the local feature layer.

$$E_i = GMP(Map(F_i, B)) \times S_j, \tag{6}$$

where $Map(F_i, B)$ implies mapping the global feature layer F_i in conjunction with the example box coordinates B. And the GMP represents pooling different example features into a unified size. S_j is the scale factor.

$$C_i = Conv(F_i, E_i), \tag{7}$$

$$C = CON(GMP(C_i, C_{i-1})), \tag{8}$$

where $C_i = Conv(F_i, E_i)$ refers that we using the layer E_i as the convolutional layer to perform convolutional operations on layer F_i, which gets the relationship between the local features and global features. CON represents the concatenating information from different feature layers.

$$Y = Adap(Reg(C), flag), \tag{9}$$

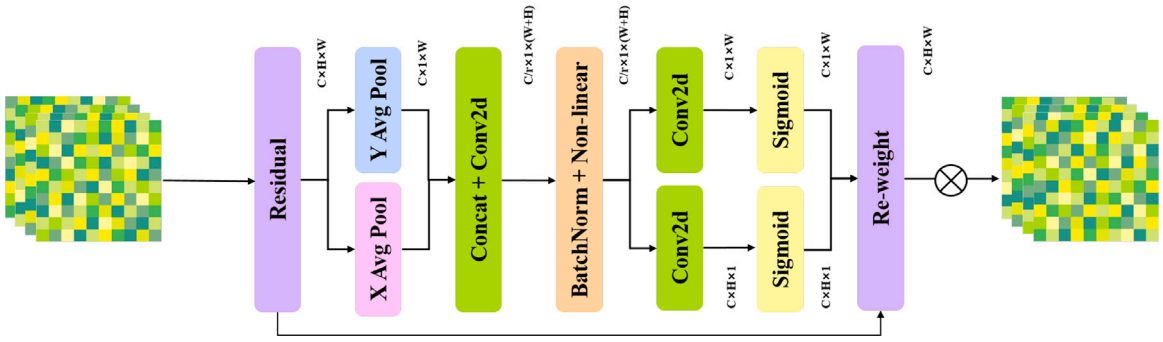


Fig. 6. Coordinate attention block.

Fig. 7. Samples from four datasets.

where $\mathcal{R}eg(C)$ represents the initial density map obtained using the regression module. $flag$ denotes whether the Ap module is used for fine-tuning parameters.

3.4. Multi-layer coordinate attention

In our network, the combination of the ResNet101FPN layer with MLCA allows for better preservation of spatial-channel feature information. This facilitates the adjustment for the subsequent MAC module and highlights regions of interest. Inspired by Multi-view learning (Ye et al., 2022), we apply Coordinate Attention (Hou et al., 2021) to the output map of each ResNet101FPN layer as shown in Fig. 6, which is motivated by its lightweight. It enhances our model's ability for cross-regional recognition and further mitigates the impact of background noise on vegetation.

Previous attention mechanisms often employ global pooling to compress the feature information. However, this often results in the loss of target positional information. In contrast, the Coordinate Attention (CA) mechanism encodes features along a single dimension, effectively preserving the key spatial information. The specific process is as follows:

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_c(i, j), \tag{10}$$

where $x_c(i, j)$ represents the two-dimensional feature map of the channel c, with (i, j) as coordinates. H is the height of the feature map, W is the width of the feature map, and z_c represents the generated channel-encoded feature.

Given the input feature x, we encode each channel along the horizontal and vertical directions using pooling kernels of size $(H, 1)$ and $(1, W)$ respectively, which yields two directional feature maps. Utilizing the generated representation information, we reduce the convolutional dimension as described in Eq. (11):

$$f = \sigma(F_1([z_h, z_w])), \tag{11}$$

where z_h and z_w represent the features of the previously obtained channel information in different dimensions. F_1 denotes the convolutional dimension reduction. σ represents the Batch Normalization and non-linear operations.

Finally, the concatenated features are split back into two-dimensional features. Convolutional transformations are applied to increase the channel dimension. The process ensures channel information while preserving positional information and capturing regions of interest. The formula is as follows:

$$g^h = \delta(F_h(f^h)), \tag{12}$$

$$g^w = \delta(F_w(f^w)), \tag{13}$$

$$y_c(i, j) = x_c(i, j) \times g_c^h(i) \times g_c^w(j), \tag{14}$$

where f^h and f^w represent the channel dimension reduction features in the height and width respectively. F_h and F_w indicate convolutional operations for dimension expansion. δ represents the Sigmoid operation. g^h and g^w are the weight vectors obtained for different dimensions. y_c represents the obtained final feature maps.

3.5. Mapping and correlation structure

This module aims to obtain relevant features as shown in Fig. 4., and its operation can be divided into two steps. The first step involves mapping the input global feature layer F_i with the bounding box coordinates B to obtain feature layer E_i. The Mapping operation is to extract features from the specific region of interest corresponding to an individual tree. The extracted region serves as a template for the subsequent feature extraction. For the extracted sample feature maps, we generate feature map groups by undergoing multi-scale transformations. The second step entails using E_i as a convolutional layer to connect the global feature layer F_i and obtain the relevant feature layer C_i. Previous networks mostly upsampled features before merging them into global features. We believe that this approach does not adequately consider the relationship between local and global information. Therefore, we propose the Correlation operation to perform mutual convolution operations between sample and global features. By utilizing individual-tree template information, it is possible to further enhance the extraction of features from the original global features. Finally, these relevant feature layers are uniformly pooled and concatenated together. Our

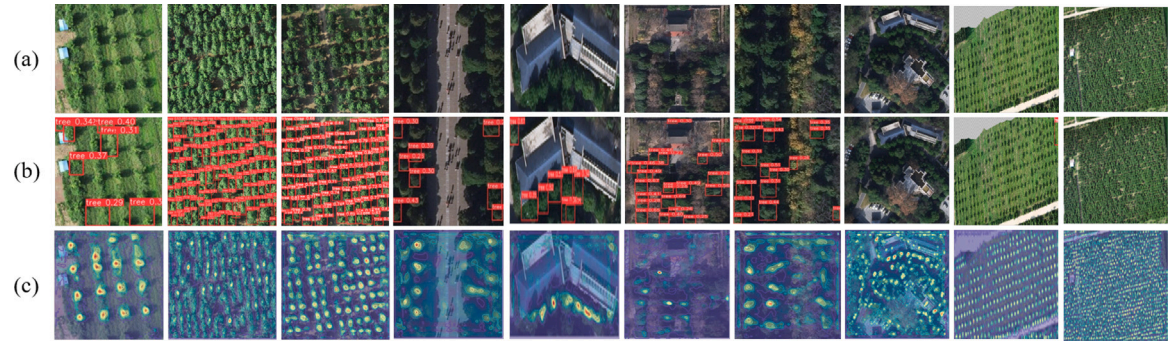


Fig. 8. Visualization comparison of VrsNet and YoloV8 in extraction on complex Vegetation Data. (a) Input images. (b) Prediction results of YoloV8. (c) Prediction results of VrsNet.

experimental evidence demonstrates that the traditional network simply concatenates local features with global features is inferior accuracy compared with the network that implicitly incorporates the correlation between local and global features.

3.6. Adaption module

The core pre-adaption involves allowing the network to learn the features of the image before making predictions. The purpose is to enable the network to better understand the presentation size of the density distribution and determine the area range. This helps in achieving more accurate predictions for segmentation and counting, especially in scenarios with varying spatial scales. Within this module, the designed "One-Trend" loss tends to make the density of the sample box region approach 1. This delineates the area of tree crowns, and subsequently, "perturbation" loss is applied to constrain the Gaussian distribution. The number of steps is elastically controllable, playing a role in fine-tuning the network parameters for the regression module. This flexibility allows for enhancing the performance of the regression module.

3.7. Loss

This section introduces four loss functions used in VrsNet, namely Mae loss, Dice loss, One-trend loss, and Perturbation loss. During training, we utilize a combination of Mae loss and Dice loss. During the Adaption module, we propose a new one-trend loss and combine it with the Perturbation loss.

***Dice loss*—** Drawing from Fausto Milletari's gradient visualization in semantic segmentation (Milletari et al., 2016), we incorporate the Dice loss. Our network aims to constrain shape differences between the predicted Y and actual G density maps, particularly in scenes with pronounced imbalances between positive and negative samples. This constraint enhances the network's ability to the intricate texture and morphology of trees as described by the following formula:

$$G^{'} = bin(Sig(G), \varnothing), \tag{15}$$

$$Y^{'} = bin(Sig(Y), \varnothing), \tag{16}$$

$$Sig(x) = \frac{1}{1+e^{-x}}, \tag{17}$$

where Sig represents the mapping of the density map using the Sigmoid function, we then set a threshold $\varnothing = 0.5$ and binarize Y and G, mapping values greater than $\varnothing$ to 1 and the rest to 0. The binarized density maps are denoted as Y' and G'. Finally, by inputting these binarized maps into the Dice loss, we quantify the contour similarity between the two maps. The Dice loss can be described as:

$$\mathcal{L}_{Dice} = 1 - \frac{2\left|G' \bigcap Y'\right|}{|G'| + |Y'|}, \tag{18}$$

Table 1
The kernels of generating ground truth.

Datasets	Generating method
TreeFsc Fsc-147 ShanghaiTech_A	Geometry-adaptive kernels
Carpk ShanghaiTech_B	Fixed kernel:10 Fixed kernel:15

***MAE loss*—** MAE (Mean Absolute Error) is a metric that measures the difference between predicted values and actual observations. It calculates the average of the prediction errors for each individual. The formula is:

$$\mathcal{L}_{Mae} = \frac{1}{n}\sum_{i=1}^{n}|Y_i - G_i|, \tag{19}$$

Therefore, the loss during the training phase is described as:

$$\mathcal{L}_{train} = \alpha_1 \mathcal{L}_{Mae} + \alpha_2 \mathcal{L}_{Dice}, \tag{20}$$

where α_1 and α_2 are scalar hyper-parameters, and the learning rate is 10^{-5}. In our testing, recommended values are $\alpha_1 = 0.1$ and $\alpha_2 = 0.00005$. This ensures that both losses are maintained at roughly the same order of magnitude, which helps in keeping the two losses within a comparable range.

***Onetrend loss*—** Onetrend loss aims to encourage predictions within the sample region to approach 1 as closely as possible. The penalty curve is conceptualized as a quadratic curve centered at 1. This approach ensures that our penalty extends to those values greater than 1. Consequently, we select a quadratic function to mitigate loss near 1. The Min function serves to cap the upper limit, preventing gradient explosion. The formula is as follows:

$$\mathcal{L}_{O-t} = min(2, 2 \times (\left\|Y_{b_i}\right\| - 1)^2), \tag{21}$$

where Y_{bi} represents the sum of pixels in the predicted density map Y within the region b_i, $(\left\|Y_{b_i}\right\| - 1)^2$ calculates the proximity of the estimated quantity. min is the Miniaturization function, which limits the penalty strength.

***Perturbation loss*—** Perturbation loss is inspired by the tracking algorithm based on correlation filters (Ranjan et al., 2021). The predicted density map Y is the result of the correlated feature map. Essentially, the correlation response map should exhibit a Gaussian distribution for density estimation at the sample positions. Therefore, given a Gaussian density window $G_{h\times w}$, this loss can be described as:

$$\mathcal{L}_{Per} = \sum_{b_i \epsilon B}\left\|Y_{b_i} - G_{h\times w}\right\|_2^2, \tag{22}$$

where $G_{h\times w}$ represents the Gaussian kernel for the annotated image's blurred region, it serves as a way to model the uncertainty or fuzziness

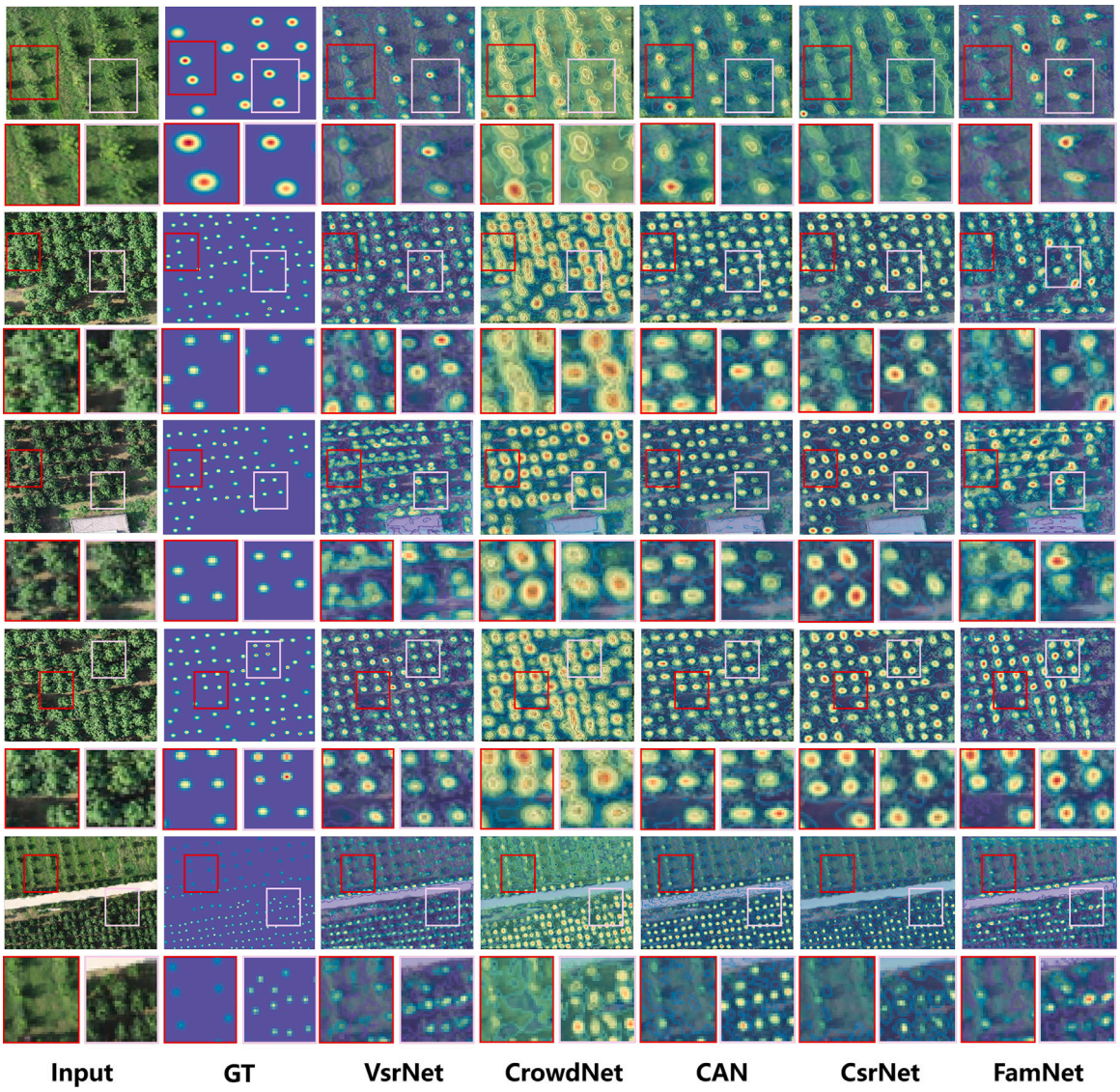


Fig. 9. Visualization of predicted density contour maps on the TreeFsc dataset, with the red and pink rectangles highlighting regions for detailed inspection. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

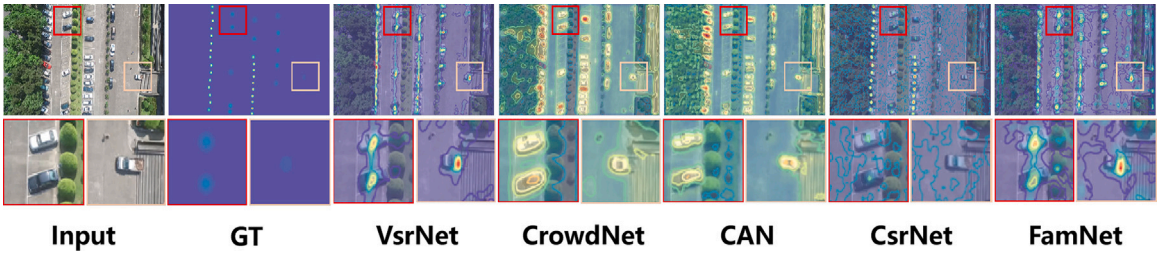


Fig. 10. Visualization of predicted density contour maps on the Carpk dataset, with the red and pink rectangles highlighting regions for detailed inspection. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

in the ground truth annotations. Therefore, the loss during the Adaption Module can be described as:

$$\mathcal{L}_{Ap} = \beta_1 \mathcal{L}_{O-t} + \beta_2 \mathcal{L}_{Per}, \tag{23}$$

where we set the default step to 200, and after experimentation, the values for β_1 and β_2 are chosen to be 10^{-8} and 10^{-4} respectively.

3.8. Algorithm complexity

Our algorithm's key module is the Mapping and Correlation module for the feature extraction. The measurement of its time complexity depends on the size of the example feature box. Our model's input is not constrained by image size or box size. The CA attention mechanism, as a lightweight attention mechanism, is not weaker than the MLCA layer constructed with ResNet101, with parameters at 55.317M. During training, we freeze the ResNet101 parameters and retain the CA

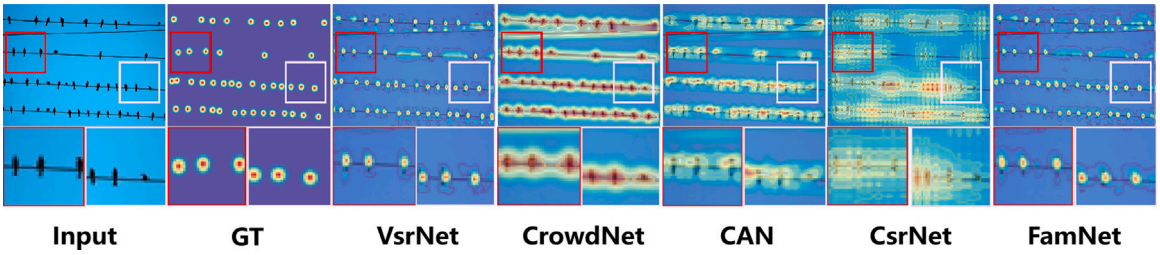


Fig. 11. Visualization of predicted density contour maps on the Fsc-147 dataset, with the red and pink rectangles highlighting regions for detailed inspection. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

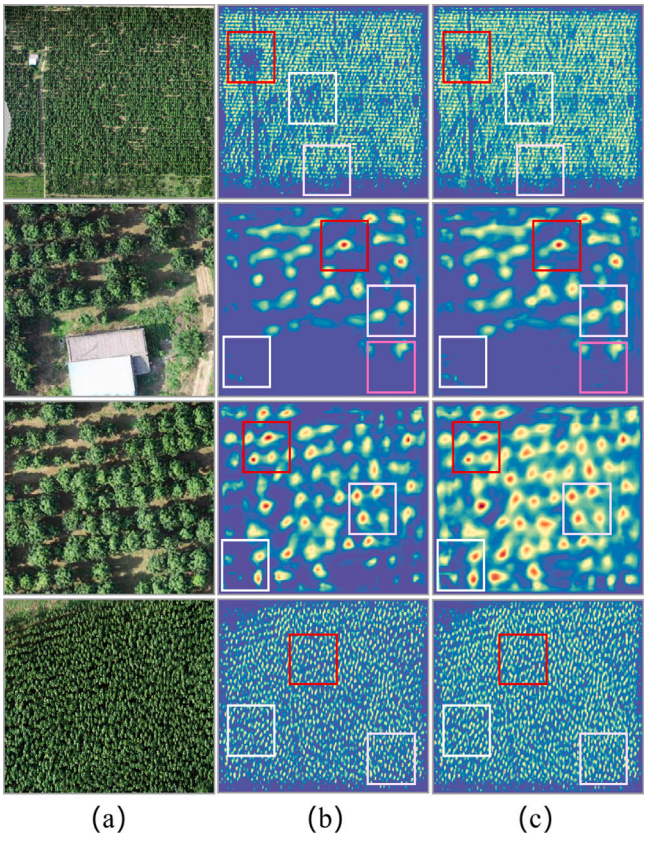


Fig. 12. Visualization of steps in the Ap module. (a) The original input data. (b) The results without using the Ap module. (c) The results with a step count of 1000. The rectangles are used to emphasize detail changes.

module parameters for feature learning. Regression is a conventional up-sampling triple regression module with 761.053K parameters. For an image with dimensions of 512 × 512 pixels, it can undergo 30 steps per second. The total adaption steps depend on the chosen parameter.

4. Experiments details

4.1. Datasets

This section overviews the four datasets utilized in the study. Samples are depicted in Fig. 7. Table 1 presents the Gaussian kernel sizes for generating ground truth. These datasets show intricate variations in scenes and categories, which further demonstrates our model's abilities in different visual tasks.

TreeFsc is the dataset we primarily investigated, focusing on individual tree detection. We conduct manual selection and segmentation to choose suitable vegetation areas for our study. The dataset encompasses a wide range of spatial scales, vegetation types, and growth environments. Drawing inspiration from the methodology outlined in Boominathan et al. (2016), we employ feature pyramid processing on the dataset at different resolutions. ***Carpk*** dataset serves as a classic dataset in the field of object detection. It further challenges the model's recognition capabilities under different lighting conditions. $\boldsymbol{Fsc-147}$ is a challenging dataset for few-shot multi-class density detection, spanning 147 different types of objects with rich environmental variations. It further tests the model's cross-class density recognition capabilities. ***ShanghaiTech*** is used for crowd detection, which is divided into Part A and Part B. It can evaluate the model's density counting capabilities.

4.2. Evaluation metrics

We evaluate the performance and effectiveness of models using the following metrics:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|C_i - \hat{C}_i\right|, \tag{24}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(C_i - \hat{C}_i)^2}, \tag{25}$$

where MAE (Mean Absolute Error) and RMSE (Root Mean Square Error) serve as effective evaluation metrics for density detection models. MAE provides a straightforward measure of average absolute errors. RMSE accounts for the error magnitude and direction.

4.3. Implement details

We implement the proposed network on PyTorch using an NVIDIA 3080 GPU (12 GB). Our backbone network is ResNet101, and we utilize pre-trained weights from ImageNet. Our training batch size is 1. All models are trained for 300 epochs on different datasets, employing the Adam optimizer with a learning rate set to 0.00001.

5. Experiment results and analysis

This section demonstrates the performance of our network through quantitative comparisons and visualizations, providing both numerical metrics and visual quality of our effectiveness.

5.1. Comparison with state-of-the-art methods

We compare the proposed VrsNet with density detection methods, including the single-branch, multi-column branch, and specially constructed architectures. In quantitative analysis and visual comparisons, we demonstrate our advantages in comparison with CrowdNet (Boominathan et al., 2016), Mcnn (Zhang et al., 2016), CsrNet (Li et al.,

Table 2
Comparison of results on TreeFsc and Fsc-147.

Methods	TreeFsc-Val		TreeFsc-Test		Fsc-147-Val		Fsc-147-Test	
Metrics	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
CANnet	14.03	17.43	13.82	18.97	33.40	87.56	31.19	83.20
CsrNet-b	15.12	20.03	16.24	21.52	36.45	93.12	34.29	87.52
CrowdNet	29.64	35.15	28.93	37.32	49.87	107.42	48.32	98.43
Mcnn	32.97	45.18	35.83	48.24	60.12	124.89	58.43	116.39
FamNet-	17.63	28.12	16.49	30.12	–	–	–	–
SwitchNet	26.63	40.18	28.43	41.25	56.84	112.54	54.83	109.85
FamNet+	12.47	16.90	13.15	17.03	**24.02** ↓	**70.01** ↓	**23.12** ↓	**68.76** ↓
VrsNet(ours)	**9.02** ↓	**12.15** ↓	**9.67** ↓	**12.82** ↓	28.54	80.12	26.30	65.41

Table 3
Comparison of results on Carpk and ShanghaiTech.

Methods	Carpk-Val		Carpk-Test		ShanghaiTech_B-Test		ShanghaiTech_A-Test	
Metrics	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
CANnet	9.01	13.63	8.82	13.91	**7.9** ↓	**12.3** ↓	**62.43** ↓	**100.15** ↓
CsrNet-b	11.52	15.80	12.61	23.42	10.61	14.97	68.29	115.42
CrowdNet	23.14	36.80	26.93	42.30	25.83	37.21	88.14	129.67
Mcnn	26.89	42.14	32.14	50.18	27.98	42.34	110.24	174.82
FamNet-	15.63	24.61	19.36	37.56	19.42	29.42	–	–
SwitchNet	24.63	39.24	27.98	40.15	23.42	34.61	90.40	136.21
FamNet+	10.21	17.53	12.55	18.03	12.53	19.47	74.15	124.58
VrsNet(ours)	**8.04** ↓	**11.43** ↓	**8.64** ↓	**11.71** ↓	8.28	13.84	70.20	120.15

Table 4
Ablation study.

BackBone	FPN-3	FPN-4	Multi-scale layer CA	Ap	Loss	MAE	RMSE
ResNet50	✓					19.14	28.27
ResNet50		✓	✓			14.92	23.41
ResNet50		✓	✓	100		12.45	18.63
ResNet50		✓	✓	200	✓	11.19	16.64
ResNet101		✓	✓	500	✓	9.02	12.15

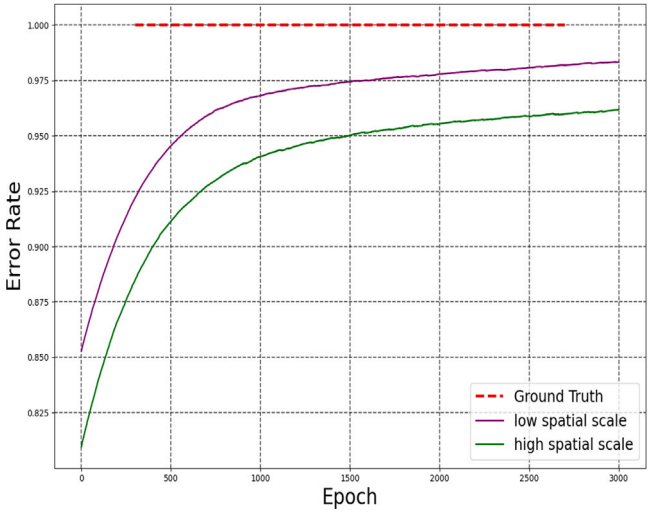


Fig. 13. Comparison of different spatial resolutions in the Ap module.

2018), CAN (Liu et al., 2019), FamNet (Ranjan et al., 2021), and SwichNet (Khoo and Ying, 2019).

By incorporating the correlation between local and global features, our approach enhances density predictions. Table 2 displays our performance on the TreeFsc and FSC-147 datasets. Specifically, our model achieves MAE and RMSE of 9.02 and 12.15 on the validation set of TreeFsc, and 9.67 and 12.82 on the test set. Simultaneously, on the FSC-147 dataset, our model achieves scores of 28.54 and 80.12 on the validation set, and 26.30 and 65.41 on the test set.

Table 5
Quantitative analysis of local example boxes B.

Number of example boxes	MAE	RMSE
Fewer(<3)	10.29	15.32
More(≥3)	9.02	12.15

In detection tasks within parking lots and urban scenes, our network has demonstrated commendable performance as shown in Table 3. On the Carpk set, our model achieves MAE and RMSE of 8.04 and 11.43, respectively. Additionally, on the ShanghaiTech dataset, our model attains the second position, following the CAN. These results demonstrate our model's superior robustness across different scenarios.

Furthermore, we randomly sample 50 images from the TreeFsc and compare the actual counting performance with various detection methods as shown in Fig. 14. The results indicate that our model's predictions better align with the ground truth images.

5.2. Visual comparative analysis across diverse datasets

This section employs visual comparative analysis to illustrate different density detection results. Additionally, we compare VrsNet with the classic box-based network YOLOv8 on vegetation with multiple scenes, multiple species and multiple spatial scales to further demonstrate our performance.

We conduct a visual comparison on the TreeFsc dataset as shown in Fig. 9. The density contour maps predicted by our model closely match the contours of the Ground Truth. This further validates the performance of our model in individual tree detection and group-tree counting. Additionally, our method identifies saplings that were not annotated in the GT images, indicating that our model has learned the connections between relevant features. We also conduct visual comparisons on the Carpk dataset as shown in Fig. 10 and the Fsc-147 dataset as shown in Fig. 11. Results indicate that our model exhibits high transferability for detecting specific categories or objects. This demonstrates that the connections between local and global features in our model facilitate cross-category detection learning.

We pre-train YoloV8 on the TreeFsc dataset and select challenging vegetation images with various scenes, species, and spatial scales to compare the cross-abilities of models as shown in Fig. 8. The results indicate that YoloV8 captures relatively clear and fixed-size vegetation

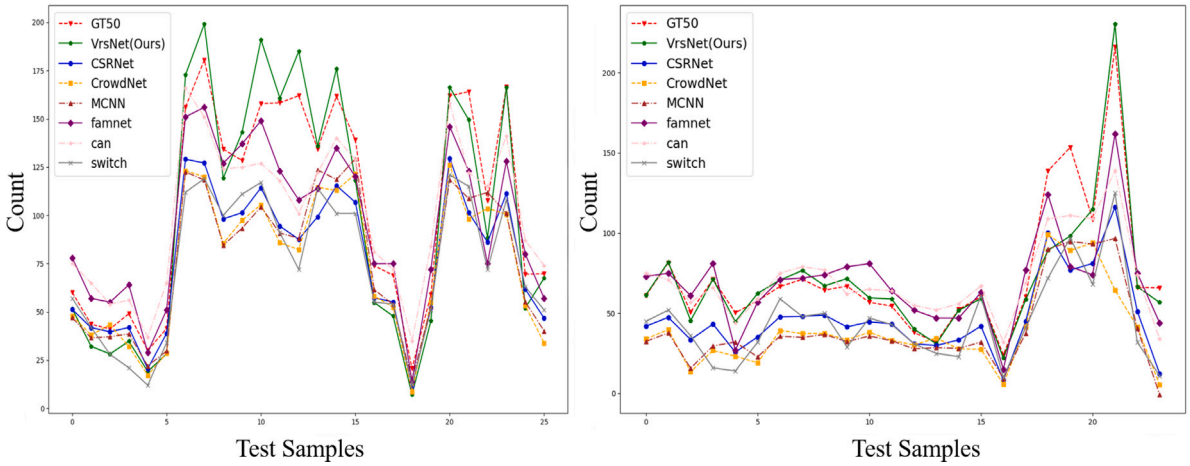


Fig. 14. Quantitative comparison of various density detection methods in randomly sampled vegetation images across different scenes.

information. However, as the scene becomes complex, YoloV8 fails to keep the accuracy. This phenomenon mainly stems from the fact that the receptive fields of traditional object detection networks are mostly fixed. When their convolutional parameters adapt to one type of scene, more data learning is needed to transition to another scene. However, our MAC module does not require training of learnable parameters. It adapts to detection tasks in any scene through examples and global feature correlated response maps.

5.3. Ablation study

To demonstrate the effectiveness of each module, we conduct ablation experiments. Firstly, we compared the different modules on the TreeFsc dataset as shown in Table 4. "BackBone" represents the main feature extraction network. "FPN-3" denotes using the third stage of ResNet as our global feature layer. "FPN-4" indicates using the feature layers from stages three and four of ResNet sequentially into the MLCA module. "Multi-scale layer CA" indicates whether we employed the MLCA module. "Ap" represents whether we applied parameter correction to the regression results, and the numerical values indicating step sizes. "Loss" means whether we trained the network using Dice loss function and One-trend loss function. Table 4 demonstrates that our model exhibits significant advantages in performance when selecting deeper skeletons and incorporating more layers for feature fusion, with an MAE of 9.02 and RMSE of 12.15. Additionally, we observe that the Ap module corrects prediction errors. The two proposed variations have also played a positive role in model convergence. Table 5 demonstrates that a greater number of local example boxes B helps contribute to the model learning relevant features.

Additionally, we conduct ablation experiments on four vegetation remote sensing images using the Ap module. Our images are divided into two different spatial scales. We perform quantitative analysis and visualizations to further validate the performance. As the epoch of the Ap module increases, our model captures more vegetation details as visualized in Fig. 12. This contributes to the enhancement of the model's ability for cross-region, cross-scene, and cross-scale recognition. Furthermore, we evaluate whether the Ap module assists in our vegetation counting using the Error Rate defined as follows:

$$ErrorRate = \frac{Pre}{GT}, \tag{26}$$

where Pre represents the estimated results and GT represents the ground truth.

To evaluate the function of the Ap module, we conduct 3000 epochs. In Fig. 13 we observe that the Ap module aids the model in adapting to the distribution of vegetation density. This adaption enables the model to achieve the instance number more effectively.

6. Application

This section utilizes the proposed model trained on our dataset for practical application on Jiaozuo, China ($34°53'59.9905''N$, $113°09'00.0057''E$), which can be divided into two parts.

The first part focuses on individual tree crown detection and segmentation. The crown segmentation demonstrates that our robustness is high in the real scenes by focusing on leaf analysis (Fan et al., 2022; Zheng et al., 2023; Pu et al., 2022). Our method employs a hierarchical segmentation approach using contour density maps. We generate a contour of constructing isolines from two-dimensional scalar fields and we partition the input space into grid cells based on the two-dimensional density distribution predicted by the VrsNet. We analyze the scalar field values at each cell to determine the contour shapes. The level parameter was set to 10. Subsequently, we apply masking to the generated images based on the contour levels. In the generated set of contour density maps, different segments can be obtained through threshold filtering at various levels as presented in Fig. 16. By combining the semantic information using the density contour maps, the segmentation results can effectively fit the contour of trees at the crown centers.

The second part involves counting individual trees (Sun et al., 2022) from various dense scenes. Our testing targets encompass a range of scene changes at large spatial scales. We compare three different natural vegetation areas. We measure the natural density of vegetation by the ratio of the number of trees to the land area (per hectare). The densities in the three different areas are approximately 400, 215, and 115 trees per hectare, respectively. From the perspective of image processing, we can reflect different vegetation image densities by the number of trees' ground truth (GT) in the images. They are as follows: 2582, 769, and 241 trees, respectively. These images under different scenarios collectively test the model's compatibility in density detection. Given the abundant vegetation resources, traditional methods for tree resource conservation and census are time-consuming and labor-intensive. By performing a comprehensive integration and summation of the density map, we obtain the total count of trees as shown in Fig. 15.

In summary, our model reduces the annotation workloads and achieves accurate dynamic results in individual tree crown segmentation. Additionally, our model improves the accuracy of the high-density individual tree-counting task. The two applications are of great value related to individual tree research.

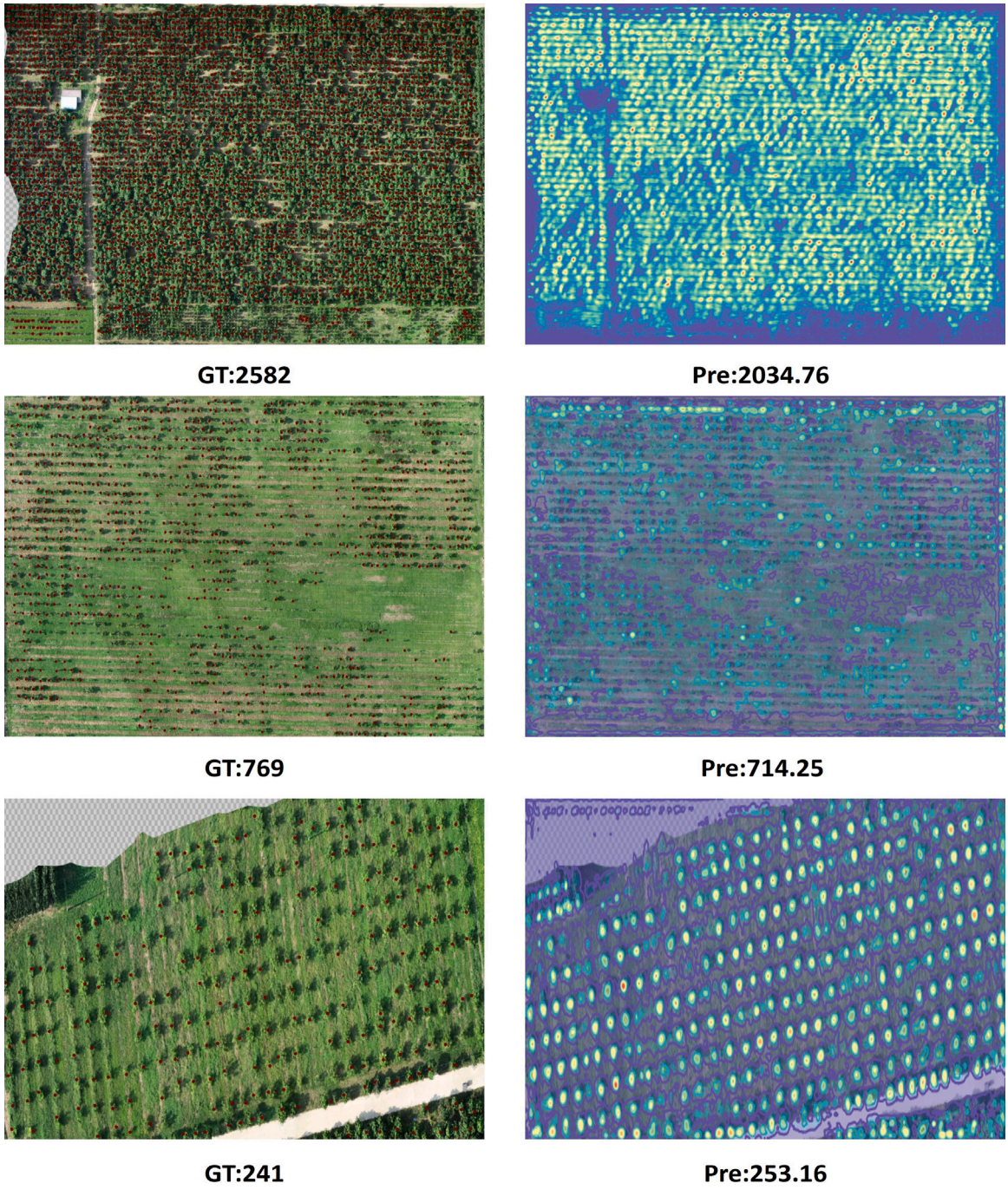


Fig. 15. Visualization of the tree counting results at different densities and scales.

7. Conclusions

This paper introduces a novel semi-supervised deep learning network for vegetation density prediction. The innovation lies in its feature extraction through the MLCA (Multi-layer Coordinate Attention module), MAC (Mapping and correlation module) and Ap (Adaption module) compared with traditional networks. The MLCA module is employed for multi-scale attention learning, enabling the constructed feature pyramid to highlight regions of interest. The MAC module emphasizes acquiring correlated features of local and global features rather than the fusion process, which helps the model learn more semantically informative connections. Simultaneously, the Ap module assists the model in adapting to the vegetation differences. In quantitative analysis of tree detection, the proposed method demonstrates superior performance, with a 27.6% increase in MAE and a 28.1% increase in RMSE. Furthermore, our work considers the semantic contour information of Gaussian distributions implicitly contained in density maps, and we propose three practical downstream applications by the proposed network: individual tree segmentation, integral counting and individual tree detection. Our performance is tested on complex large-scale vegetation datasets with varying scales and achieves high visual quality and accuracy.

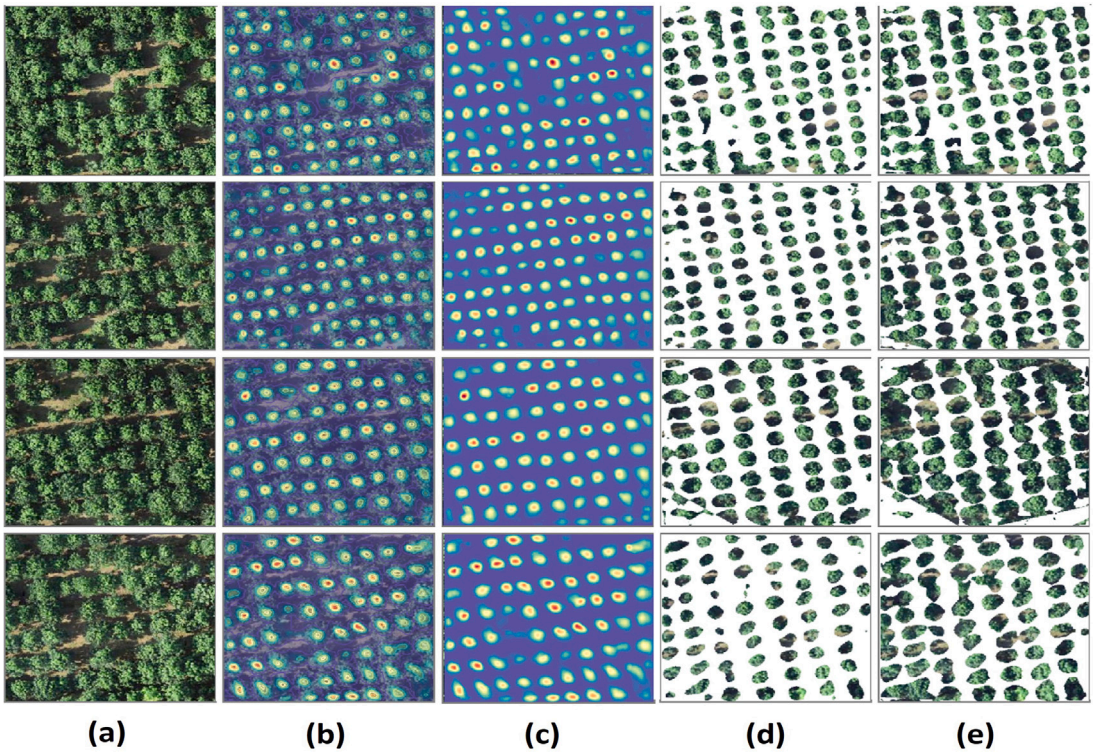


Fig. 16. Visualization of individual tree detection and segmentation results. (a) The input image. (b) The contour density map. (c) The filled contour density map. (d) The segmentation results with higher threshold. (e) The segmentation results with lower threshold.

CRediT authorship contribution statement

Taige Luo: Writing – original draft, Methodology. **Wei Gao:** Writing – original draft, Formal analysis. **Alexei Belotserkovsky:** Validation, Investigation. **Alexander Nedzved:** Validation, Resources. **Weijie Deng:** Visualization, Software. **Qiaolin Ye:** Validation, Resources. **Liyong Fu:** Investigation, Funding acquisition. **Qiao Chen:** Data curation, Formal analysis. **Wenjun Ma:** Validation, Writing – original draft. **Sheng Xu:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that there are no conflict of interests, we do not have any possible conflicts of interest.

Data availability

Data will be made available on request.

Acknowledgments

This research is supported in part by the Fundamental Research Funds for the Central Nonprofit Research Institution of CAF (CAFYBB2022ZB002), and in party by National Natural Science Foundation of China (NO. 62102184, NO. 32371877)

References

Boominathan, L., Kruthiventi, S.S.S., Babu, R.V., 2016. CrowdNet: A deep convolutional network for dense crowd counting. In: MM'16: Proceedings of the 2016 ACM Multimedia Conference. pp. 640–644. http://dx.doi.org/10.1145/2964284.2967300.

Cheng, Y., Lan, S., Fan, X., Tjahjadi, T., Jin, S., Cao, L., 2023. A dual-branch weakly supervised learning based network for accurate mapping of woody vegetation from remote sensing images. Int. J. Appl. Earth Obs. Geoinf. 124, 103499. http://dx.doi.org/10.1016/j.jag.2023.103499.

Fan, X., Luo, P., Mu, Y., Zhou, R., Tjahjadi, T., Ren, Y., 2022. Leaf image based plant disease identification using transfer learning and feature fusion. Comput. Electron. Agric. 196, 106892. http://dx.doi.org/10.1016/j.compag.2022.106892.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 580–587. http://dx.doi.org/10.1109/CVPR.2014.81.

Guo, Y., Wu, C., Du, B., Zhang, L., 2022. Density Map-based vehicle counting in remote sensing images with limited resolution. ISPRS J. Photogramm. Remote Sens. 189, 201–217. http://dx.doi.org/10.1016/j.isprsjprs.2022.05.004.

He, K., Gkioxari, G., Dollar, P., Girshick, R., 2017. Mask R-CNN. In: 2017 IEEE International Conference on Computer Vision. ICCV, pp. 2980–2988. http://dx.doi.org/10.1109/ICCV.2017.322.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 770–778. http://dx.doi.org/10.1109/CVPR.2016.90.

Hou, Q., Zhou, D., Feng, J., 2021. Coordinate attention for efficient mobile network design. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2021. pp. 13708–13717. http://dx.doi.org/10.1109/CVPR46437.2021.01350.

Hui, Z., Cheng, P., Yang, B., Zhou, G., 2022. Multi-level self-adaptive individual tree detection for coniferous forest using airborne LiDAR. Int. J. Appl. Earth Obs. Geoinf. 114, 103028. http://dx.doi.org/10.1016/j.jag.2022.103028.

Jayarao, B., Pillai, S., Sawant, A., Wolfgang, D., Hegde, N., 2004. Guidelines for monitoring bulk tank milk somatic cell and bacterial counts. J. Dairy Sci. 87 (10), 3561–3573. http://dx.doi.org/10.3168/jds.S0022-0302(04)73493-1.

Jiang, T., Liu, S., Zhang, Q., Xu, X., Sun, J., Wang, Y., 2023. Segmentation of individual trees in urban MLS point clouds using a deep learning framework based on cylindrical convolution network. Int. J. Appl. Earth Obs. Geoinf. 123, 103473. http://dx.doi.org/10.1016/j.jag.2023.103473.

Khoo, Y., Ying, L., 2019. SwitchNet: A neural network model for forward and inverse scattering problems. SIAM J. Sci. Comput. 41 (5), A3182–A3201. http://dx.doi.org/10.1137/18M1222399.

Li, Y., Zhang, X., Chen, D., 2018. CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 1091–1100. http://dx.doi.org/10.1109/CVPR.2018.00120.

Liu, W., Salzmann, M., Fua, P., 2019. Context-aware crowd counting. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR 2019, pp. 5094–5103. http://dx.doi.org/10.1109/CVPR.2019.00524.

Liu, Q., Yan, C., Xiao, Q., Yan, G., Fang, L., 2012. Separating vegetation and soil temperature using airborne multiangular remote sensing image data. Int. J. Appl. Earth Obs. Geoinf. 17, 66–75. http://dx.doi.org/10.1016/j.jag.2011.10.003.

Milletari, F., Navab, N., Ahmadi, S.-A., 2016. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In: Proceedings of 2016 Fourth International Conference on 3D Vision. 3DV, pp. 565–571. http://dx.doi.org/10.1109/3DV.2016.79.

Mundhenk, T.N., Konjevod, G., Sakla, W.A., Boakye, K., 2016. A large contextual dataset for classification, detection and counting of cars with deep learning. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), Computer Vision - ECCV 2016, PT III. Vol. 9907, pp. 785–800. http://dx.doi.org/10.1007/978-3-319-46487-9_48.

Nasiri, V., Hawryło, P., Janiec, P., Socha, J., 2023. Comparing object-based and pixel-based machine learning models for tree-cutting detection with PlanetScope satellite images: Exploring model generalization. Int. J. Appl. Earth Obs. Geoinf. 125, 103555. http://dx.doi.org/10.1016/j.jag.2023.103555.

Pu, W., Di, C., Shaobo, X., Cheng, W., 2022. A crown guess and selection framework for individual tree detection from ALS point clouds. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 15, 3533–3538. http://dx.doi.org/10.1109/JSTARS.2022.3171771.

Ranasinghe, Y., Nair, N.G., Bandara, W.G.C., Patel, V.M., 2023. Diffuse-denoise-count: Accurate crowd-counting with diffusion models. arXiv:2303.12790.

Ranjan, V., Sharma, U., Nguyen, T., Hoai, M., 2021. Learning to count everything. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2021. pp. 3393–3402. http://dx.doi.org/10.1109/CVPR46437.2021.00340.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 779–788. http://dx.doi.org/10.1109/CVPR.2016.91.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 28. NIPS 2015, Vol. 28.

Sun, Y., Li, Z., He, H., Guo, L., Zhang, X., Xin, Q., 2022. Counting trees in a subtropical mega city using the instance segmentation method. Int. J. Appl. Earth Obs. Geoinf. 106, 102662. http://dx.doi.org/10.1016/j.jag.2021.102662.

Tolan, J., Yang, H.-I., Nosarzewski, B., Couairon, G., Vo, H.V., Brandt, J., Spore, J., Majumdar, S., Haziza, D., Vamaraju, J., Moutakanni, T., Bojanowski, P., Johns, T., White, B., Tiecke, T., Couprie, C., 2024. Very high resolution canopy height maps from RGB imagery using self-supervised vision transformer and convolutional decoder trained on aerial lidar. Remote Sens. Environ. 300, 113888. http://dx.doi.org/10.1016/j.rse.2023.113888, URL: https://www.sciencedirect.com/science/article/pii/S003442572300439X.

Wang, C., Zhang, H., Yang, L., Liu, S., Cao, X., 2015. Deep people counting in extremely dense crowds. In: MM'15: Proceedings of the 2015 ACM Multimedia Conference. pp. 1299–1302. http://dx.doi.org/10.1145/2733373.28063370-12345-67-8/90/01.

Xu, S., Li, X., Yang, H., Xu, S., 2023. R-ProjNet: an optimal rotated-projection neural network for wood segmentation from point clouds. Remote Sens. Lett. 14 (1), 60–69. http://dx.doi.org/10.1080/2150704X.2022.2163203.

Xu, W., Liang, D., Zheng, Y., Xie, J., Ma, Z., 2021a. Dilated-scale-aware category-attention ConvNet for multi-class object counting. IEEE Signal Process. Lett. 28, 1570–1574. http://dx.doi.org/10.1109/LSP.2021.3096119.

Xu, S., Wang, R., Wang, H., Yang, R., 2021b. Plane segmentation based on the optimal-vector-field in LiDAR point clouds. IEEE Trans. Pattern Anal. Mach. Intell. 43 (11), 3991–4007. http://dx.doi.org/10.1109/TPAMI.2020.2994935.

Xu, S., Ye, N., Xu, S., Zhu, F., 2018. A supervoxel approach to the segmentation of individual trees from LiDAR point clouds. Remote Sens. Lett. 9 (6), 515–523. http://dx.doi.org/10.1080/2150704X.2018.1444286.

Ye, Q., Huang, P., Zhang, Z., Zheng, Y., Fu, L., Yang, W., 2022. Multiview learning with robust double-sided twin SVM. IEEE Trans. Cybern. 52 (12), 12745–12758. http://dx.doi.org/10.1109/TCYB.2021.3088519.

Zang, Y., Wang, S., Guan, H., Peng, D., Chen, J., Chen, Y., Delavar, M.R., 2024. VAM-Net: Vegetation-Attentive deep network for Multi-modal fusion of visible-light and vegetation-sensitive images. Int. J. Appl. Earth Obs. Geoinf. 127, 103642. http://dx.doi.org/10.1016/j.jag.2023.103642.

Zhang, Z., Rong, J., Qi, Z., Yang, Y., Zheng, X., Gao, J., Li, W., Yuan, T., 2024. A multi-species pest recognition and counting method based on a density map in the greenhouse. Comput. Electron. Agric. 217, 108554. http://dx.doi.org/10.1016/j.compag.2023.108554.

Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y., 2016. Single-image crowd counting via multi-column convolutional neural network. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 589–597. http://dx.doi.org/10.1109/CVPR.2016.70.

Zhao, Q., Lyu, S., Zhao, H., Liu, B., Chen, L., Cheng, G., 2024. Self-training guided disentangled adaptation for cross-domain remote sensing image semantic segmentation. Int. J. Appl. Earth Obs. Geoinf. 127, 103646. http://dx.doi.org/10.1016/j.jag.2023.103646.

Zheng, J., Yuan, S., Li, W., Fu, H., Yu, L., 2023. A review of individual tree crown detection and delineation from optical remote sensing images. arXiv:2310.13481.