Improved model for detecting randomly oriented objects on remote sensing images

Ihor A. Pilkevych, Mykola P. Romanchuk, Olena M. Naumchak, Dmytro L. Fedorchuk and Leonid M. Naumchak

Korolyov Zhytomyr Military Institute, 22 Myru Ave., Zhytomyr, 10004, Ukraine

Abstract

Object detection in optical remote sensing images is an important task. In recent years, methods based on convolutional neural networks have shown progress. However, due to object variations such as scale, aspect ratio, and random orientation, detection is difficult to further improve. Most convolutional neural networks use rectangular bounding boxes for object detection parallel to the image coordinate axes, which is effective. However, for military objects in satellite images, which may have a large aspect ratio and be randomly oriented, rectangular bounding boxes may not always provide sufficient target localization. In this paper, methods based on the rotation of rectangular frames or other polygonal boundaries are considered, including the following. Rotation Region Proposal Network (RRPN) and Rotation Region CNN (R2CNN). One-stage models such as SSD, YOLO, and RetinaNet have demonstrated high speed and accuracy. The new YOLOv11 model, which is a further development of the one-stage model approaches, demonstrates an increase in the accuracy and speed of object detection and recognition. The purpose of the study is the analysis of modern neural network models and their improvement to enhance the accuracy of detecting and recognizing small densely located, randomly oriented objects on satellite images. The paper proposes a model with a five-parameter regression that includes the parameter of the rotation angle of the bounding box. The results of the study show that this model improves the accuracy of object detection in complex scenarios by providing accurate determination of their orientation and scale.

Keywords

remote sensing, randomly oriented object, detector, object detection

1. Introduction

In world practice, computer vision technologies are widely used to process remote sensing images. To identify objects in remote sensing images, it is necessary to solve the tasks of detecting, recognizing, assigning accurate bounding boxes or masks for small, randomly oriented objects, separating them from the background, and providing object class labels [1, 2].

Currently, a large number of models based on convolutional neural networks have been developed to improve the accuracy of object detection and recognition. In the process of recognizing and locating an object, the neural network model uses a rectangular bounding box to detect it, and then classifies and distinguishes between the object itself or the background within it [3, 4]. Most cases of object detection from a perspective parallel to the Earth are parallel to the image coordinate axis with a small aspect ratio. As a result, a rectangular bounding box can better cover objects and contain less background [5]. However, in the case of observing military objects with a large aspect ratio and disordered direction in images acquired remotely from an observation angle perpendicular to the Earth [6], it is not possible to accurately surround the object with a rectangular bounding box alone [7].

doors-2025: 5th Edge Computing Workshop, April 4, 2025, Zhytomyr, Ukraine

igor.pilkevich@meta.ua (I. A. Pilkevych); romannik@ukr.net (M. P. Romanchuk); olenanau@gmail.com (O. M. Naumchak); fedor4uk.d@gmail.com (D. L. Fedorchuk); naumchak.leonid@gmail.com (L. M. Naumchak)

https://ieeexplore.ieee.org/author/37089181628 (I. A. Pilkevych); https://ieeexplore.ieee.org/author/37087013658 (M. P. Romanchuk); https://ieeexplore.ieee.org/author/37089181640 (O. M. Naumchak); https://ieeexplore.ieee.org/author/37089179622 (D. L. Fedorchuk); https://ieeexplore.ieee.org/author/37089179498 (L. M. Naumchak)

0000-0001-5064-3272 (I. A. Pilkevych); 0000-0002-0087-8994 (M. P. Romanchuk); 0000-0003-3336-1032 (O. M. Naumchak); 0000-0003-2896-3522 (D. L. Fedorchuk); 0000-0002-7311-6659 (L. M. Naumchak)

In the field of computer image processing, a detector is a model for detecting and recognizing objects. To solve the problem of detecting simple objects, one-stage and two-stage detectors are used. One-stage detectors include: SSD, YOLO, RetinaNet, R³Det, RSDet, RIDet, FCOS, CSL, DCL, GWD, KLD, KFioU, and two-stage detectors include Fast R-CNN, Faster R-CNN, Mask R-CNN, Cascade R-CNN, RRPN, R²CNN, SCRDet, SCRDet++ [8].

Classical object detection is the detection of a simple object in an image using a horizontal bounding box. Nowadays, many high-performance methods for detecting simple objects, such as the two-stage model described by Fast R-CNN [9] and Faster R-CNN [10], focus on accuracy and reduce the amount of computation to improve detection speed. To solve the problem of changing the scale of an object in an image, the pyramidal feature network (FPN) method was proposed.

Since most approaches are based on the assumption that objects are located along horizontal lines in the image, the detector uses a rectangular bounding box parallel to the coordinate axis to detect and locate the object in the image. Then it classifies the object or background directly within this frame [3, 4]. As a result, the task of detecting randomly rotated objects with a large aspect ratio arises, which increases the bounding box and, as a result, leads to overloading of the detector during classification, and in the case of detecting randomly rotated, densely spaced objects, the overlapping frames process complex scenes and make it difficult to distinguish a single object [11] (figure 1).

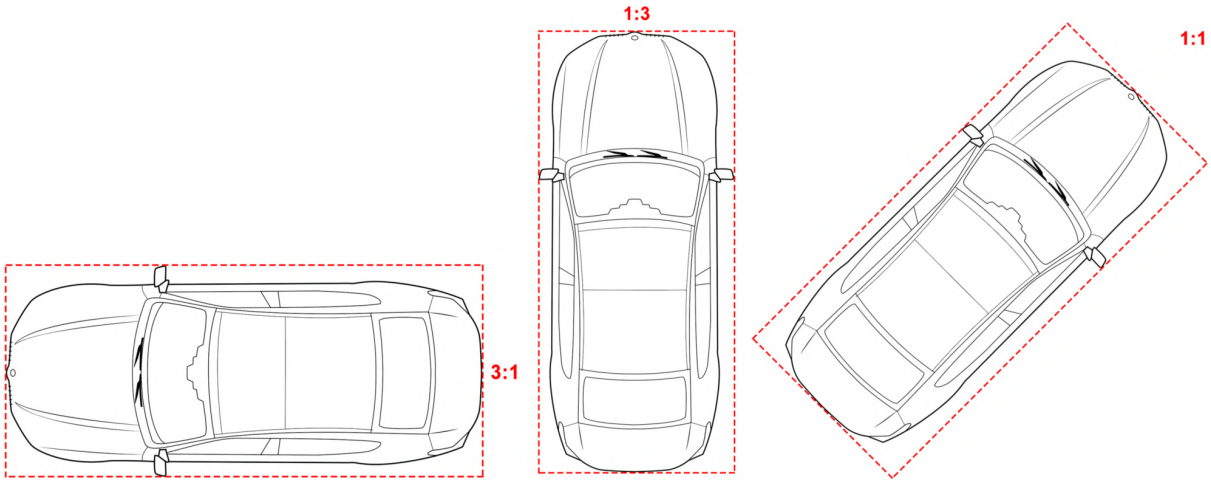

Figure 1: Traditional bounding box of an image object detector.

To solve the problem of detecting randomly oriented objects, approaches based on the rotation of a rectangular bounding box or other polygonal bounding boxes are used. For example, the Rotation Region Proposal Network (RRPN) [12] obtains a region of interest based on the rotated anchor for feature detection. The Rotational Region CNN (R2CNN) [13] is based on the Fast R-CNN, using two types of pooling size with different width-to-height ratios. However, newly developed models using the approach of two-stage detectors based on traditional horizontal region detection do not produce results with the required speed and accuracy.

The CornerNet [14], CenterNet [15], and ExtremeNet methods have gained popularity, which select and group a set of certain key points of an object, such as corners, peaks, etc., to build a bounding box.

Single-stage detection methods (single-frame multi-box SSD detector [16], YOLO family of models, and RetinaNet [17]) are based on bounding box regression. YOLOv11 [18] is the most advanced model that supports all the previous ones, and is improved by a new backbone network, detection unit, and loss function [19].

Their advantage is the higher speed of object detection and recognition. The disadvantage of the considered approaches is that they do not take into account the cases of complex scenarios on satellite images when it is necessary to detect small, densely located, randomly rotated objects, the detection of which remains relevant under such conditions.

The YOLOv11 object detection system is a single-stage system, but its accuracy is higher than most

two-stage detectors, and it is also fast. Therefore, in this paper, we use YOLOv11, on the basis of which we implement the detection of randomly rotated objects.

The purpose of the article is the analysis of neural network models and their improvement as a tool for improving the accuracy of detecting and recognizing small, arbitrarily rotated objects on satellite images.

2. Theoretical background

The considered detectors for detecting objects in images use a rectangular bounding box parallel to the coordinate axes, which, when detecting randomly rotated objects with a large aspect ratio, increases the bounding box and, as a result, leads to overloading of the detector in the case of classification. In addition, it does not provide accurate information about the object's orientation and scale.

To implement the detection of objects with randomly orientation, each detector and dataset provides its own definition of the rotation angle. The DOTA dataset [20] stores the coordinates of the four corners of the object's bounding box. R2CNN [21] uses the coordinates of the first two clockwise corners of the four $(x_1, y_1; x_2, y_2)$ and the height of the rectangle to define the frame. A common method is five-parameter regression, which adds an angle θ parameter in addition to the basic parametersxy and wh, to represent the bounding box in any direction. As shown in figure 2) (left), this is an acute angle formed by the width (or height) of the bounding box and the axis x in the range of $0 - 90°$. Another method is that the angle formed between the longest side of the rectangle and the axis x is between $-90°$ and $+90°$, as shown in figure 2) (right).

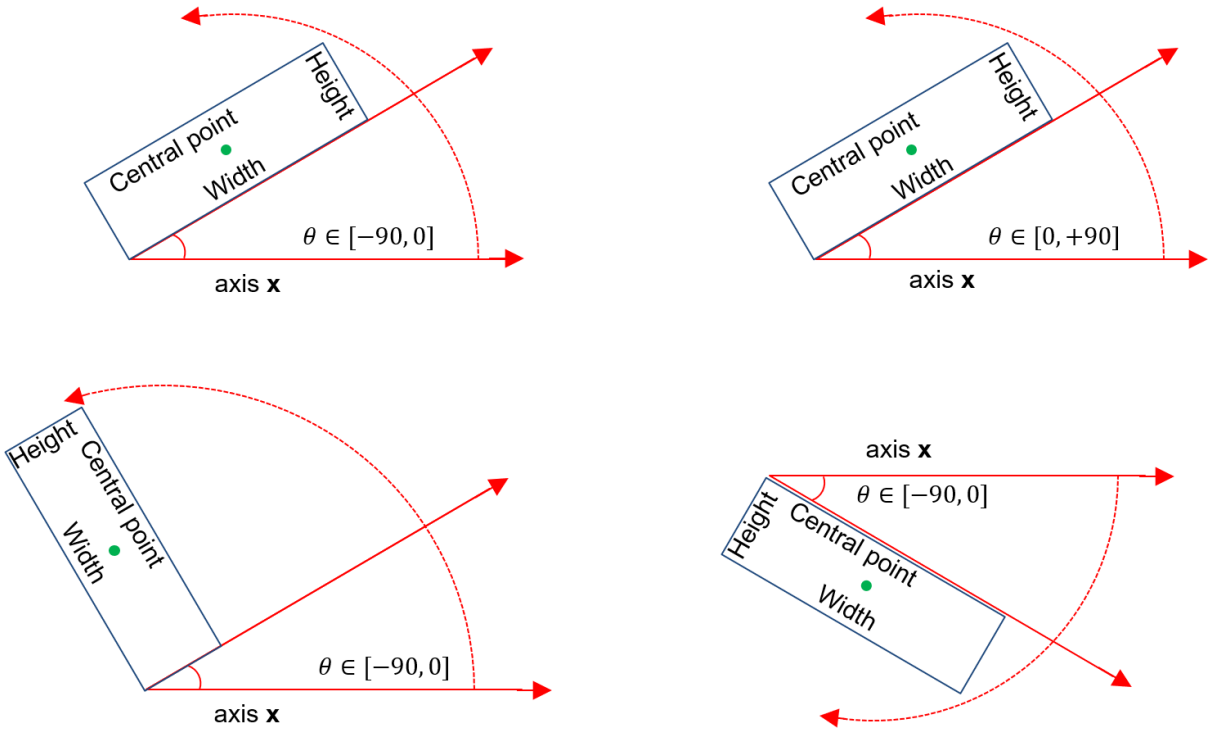

Figure 2: Defining the bounding box.

In the proposed model, the image label is pre-processed, in which the processing mainly concerns the label part, and the angle information is obtained from the spatial label $xywh$ of the object. Data preprocessing allows us to obtain the object's orientation angle from $-90°$ to $+90°$ and the division into width w and height h.

The architecture of the YOLOv11 model, which is designed for improving small object detection and accuracy while maintaining real-time inference speed, is shown in figure 4. The network consists of the following parts: input, main, and prediction. Some modules are omitted in the figure and only

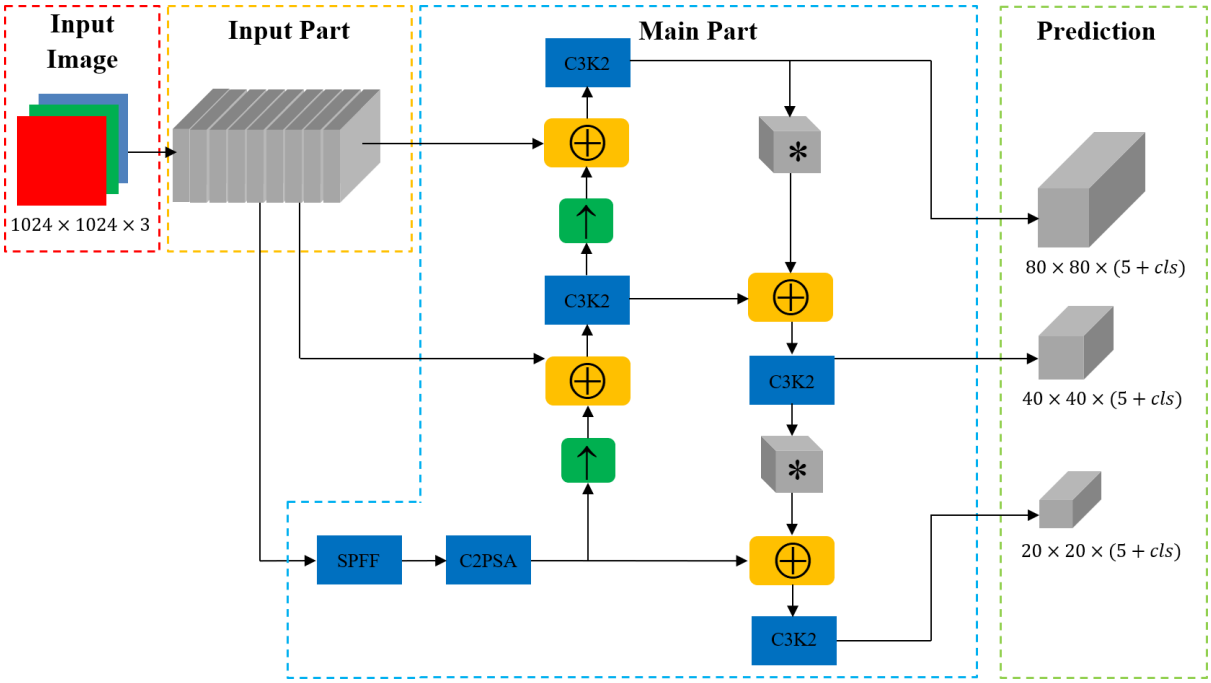

Figure 3: Model for detecting and recognizing randomly oriented objects in an image.

the general structure is shown. The input part is used to extract features from the image, from which three feature maps are sequentially extracted, which pass through the main part, where a number of operations are performed on them, such as convolution $(*)$, upsampling $(\uparrow)$ and combining $(\bigoplus)$.

The convolution $(*)$ consists of a 2D convolutional layer and a 2D batch normalization layer with SiLU activation function [22]. YOLOv11 uses C3K2 blocks to handle feature extraction at different processing stages. The C3K2 block optimizes information processing by dividing the feature map and applying a series of smaller kernel convolutions 3×3, which are faster and cheaper to compute compared to large kernel convolutions. It consists of convolution blocks at the beginning and end, followed by a series of convolution blocks with interval pooling that disregards residuals when negative, and ends with a pooling and simple convolution block.

A special feature of YOLOv11 is the use of fast spatial pyramid fusion (SPFF), which was developed to combine features from different regions of the image at different scales. To merge features, SPFF uses multiple maximal pooling operations (with different kernel sizes) to aggregate multi-scale contextual information. This improves the processing of fine-grained objects in images.

One of the significant innovations in YOLOv11 is the addition of the Cross Stage Partial with Spatial Attention (C2PSA) block. This block introduces attention mechanisms that improve the model's focus on important areas of the image, such as smaller or partially covered objects, by emphasizing spatial relevance in feature maps.

Prediction produces detection blocks for three different scales (low, medium, high) using the feature maps created by the previous processing steps. This approach ensures that small objects are detected in greater detail while larger objects are captured by higher-level features.

As a result of processing, the neural network produces three predictions for the scales $80 \times 80, 40 \times 40, 20 \times 20$. The format of the predicted object label for all scales is provided in the following format: cls – general label category and five parameters of the bounding box (x, y – coordinates of the lower left corner; w, h – width and height; θ – angle of inclination to the x-axis).

In the proposed five-parameter model (x, y, w, h, θ), regression is used to predict the rotation of the object bounding box, since weapons and military equipment samples on satellite images have a fixed aspect ratio, and the direction parallel to the longer side is defined as the direction of the object's movement. Therefore, to facilitate the regression task, the longer side is defined as w, and the shorter

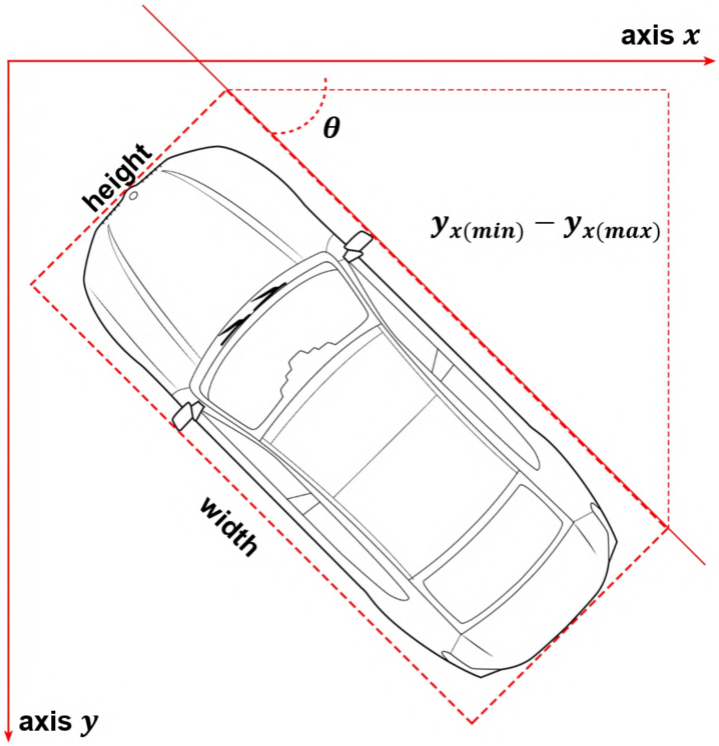

Figure 4: Calculating the angle of rotation.

side is defined as h. Thus, the direction parallel to is the direction of motion of the object. The angle between the longer side w and the axis x is the angle of rotation. Given that the required range of angles is $[-90°, 90°]$, the function $arcsin$ is chosen to calculate the angle θ. The rotation angle is calculated using the expression (figure 4):

$$\theta = \arcsin[(y_{x(min)} - y_{x(max)})/w], \tag{1}$$

The two points on the longest side $w, y_{x(min)}$, describe the value of the point on the axis y with the smaller value on the axis $x, y_{x(max)}$, describing the opposite.

To accurately determine the angles, it is also necessary to perform a conversion between the five-parameter method and the four-point annotation $x_1y_1, x_2y_2, x_3y_3, x_4y_4$. The YOLOv11 model module, which performs data preprocessing, affine and color transformations of the image, receives four points of the object's corners as input. When recalculated, the final result of target detection is the coordinates of the four corner points with a rotating bounding box applied to the original image. An example of coordinate calculation for the coordinate x_i:

$$Fx_i = \frac{(-1)^{L(Ox_i,C_x)} w cos\theta - (-1)^{L(Oy_iC_y)} h sin\theta}{2} + Cx, \tag{2}$$

$$L(ox, Cx) = \begin{cases} 0 & Ox > Cx \\ 1 & Ox < Cx \end{cases}, \tag{3}$$

where Fx_i is the final value of the point after the transformation; $L(Ox_i, C_x)$ – is the relative value of the location of the initial corner point Ox and the center point Cx.

The angle value is added to solve the problem of regressing the object's direction of rotation. For example, in a neural network, when processing an input image with 80 object detection and recognition categories and using the four-parameter method (x, y, w, h) to locate the target, the final output matrix is $F \times F \times (80 + 4 + 1)$. F provides the dimension of the feature map output by the last prediction layer, and is the probability that a certain pixel in the feature map is the center point of the object; the

main part module is located between the two layers mentioned above and provides some modules such as FPN. Therefore, to use the five-parameter positioning method, an additional channel is added to the main part to predict the angle value (figure 4).

When using the five-parameter positioning method, the center point of the object in the classification and positioning prediction matrix in the original layer of the feature map is placed in a rectangular coordinate system. As a result, during the training of the neural network model, a significant distance between the training sample and the object prediction can lead to large values of the loss function, which will not contribute to the convergence of the neural network model.

Therefore, first, the cell of the coordinate grid where the label is placed is determined. Its upper left corner is the origin. Subsequently, the coordinates xy are calculated as the offset of t_x and t_y relative to the upper left corner in the range of values $[0; -1]$, which reduces the value of the loss function. When training the neural network to increase the accuracy of localization of positive label predictions, YOLOv11 uses one training sample to create three positive predictions, which leads to a change in the range of coordinates t_x and $t_y[-0,5; 1,5]$ (figure 5).

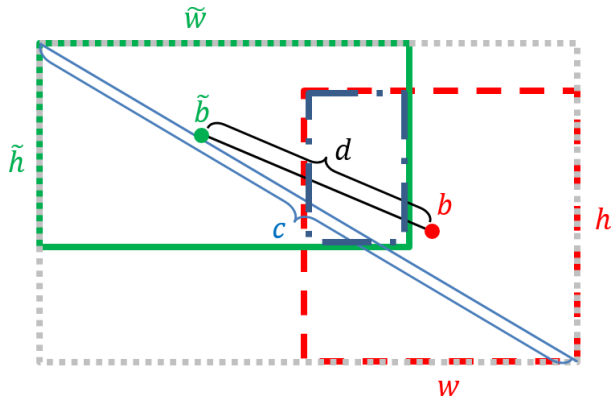

Figure 5: CIOU loss function.

The result given by the prediction part of the neural network model cannot be directly calculated for the loss function. To limit it within a given range, we use coordinate regression functions b_x, b_y, – $[-0,5; 0,5]$:

$$b_x = \frac{2}{1+e^{-t}} - 0,5 + C_x, \tag{4}$$

where b_x is the actual position of the center point of the predicted bounding box; t_x is the output value of the neural network model after calculation; C_x is the value of the grid origin; angle of inclination b_θ – $[-1,5; 1,5]$ – (calculated in radians):

$$b_\theta = \frac{3}{1+e^{-\theta}} - 1,5, \tag{5}$$

The loss functions L for training the neural network model for positioning and orienting the bounding box are:

$$L = L_{ciou} + L_{angle}, \tag{6}$$

where the loss functions L_{ciou} are for calculating the size and location of the center, and L_{angle} – for calculating the angle of rotation.

The function L_{ciou} [20] (figure 5) works with the width w, height h, distances d between the two center points of the bounding boxes and c – between the outer corners of their union. In figure 5, the bounding box of the training sample $\widetilde{B}$ is marked with a solid line, the predicted one B with a dashed line, the intersection $I(\widetilde{B}, B)$ with a dashed line, and the union $U(\widetilde{B}, B)$ with a dotted line.

The full loss function L_ciou can be described as follows:

$$L_ciou = 1 - IOU + \frac{\rho^2(\widetilde{b}, b)}{c^2} + \alpha\nu, \tag{7}$$

where ρ is the Euclidean distance between the center points $\widetilde{b}, b$ of the bounding boxes $\widetilde{B}$ and B, c is the minimum diagonal distance of their union, α and ν is the penalty of the loss function for the distance between the center points and the aspect ratio of the bounding boxes.

The components of L_ciou take into account the following:

IOU calculates the intersection area over the union of the training sample bounding box and the object prediction:

$$IOU = \frac{I(\widetilde{B}, B)}{U(\widetilde{B}, B)}, \tag{8}$$

α takes into account the aspect ratio:

$$\alpha = \frac{\nu}{(1 - IOU) + \nu}, \tag{9}$$

ν is used to measure the consistency of the aspect ratio:

$$\nu = \frac{4}{\pi^2}(\arctan\frac{\widetilde{w}}{\widetilde{h}} - \arctan\frac{w}{h})^2, \tag{10}$$

The rotation angle is calculated by the individual losses of $SmoothL1$:

$$L_{SmoothL1} = \begin{cases} 0.5(\widetilde{\theta} - \theta)^2 & \left|\widetilde{\theta} - \theta\right| < 1 \\ \left|\widetilde{\theta} - \theta\right| - 0.5 & \left|\widetilde{\theta} - \theta\right| \geq 1 \end{cases}, \tag{11}$$

where $\widetilde{\theta}$ – rotation angle of the training sample $\widetilde{B}$; θ – angle according to the forecast.

Backpropagation will gradually reduce the training losses of the neural network model to achieve the expected object detection result.

3. Experimental results

To evaluate the results of the proposed rotating detector model, comparative experiments were conducted on the DOTA reference dataset. The DOTA images were collected from Google Earth, GF-2 and JL-1 satellite remote sensing data provided by the China Satellite Data Resource and Application Center, and aerial photographs from CycloMedia. DOTA consists of RGB and grayscale images. RGB images are taken from Google Earth and CycloMedia, and grayscale images are taken from the panchromatic range of GF-2 and JL-1 satellite images. All images are saved in png format. The dataset contains 11268 remote sensing images (whose sizes vary from 800×800 to 20000×20000 pixels) with 1793658 instances, which are divided into 18 categories. Dataset composition: 4622 images with 621973 instances are the training set; 593 images with 81048 instances are the validation set; 6053 images with 1090637 instances are the test set. Each instance is labeled as a rectangle with clockwise dots. $x_1y_1, x_2y_2, x_3y_3, x_4y_4$ Half of the images in this set were used as a training set, one third as a test set, and one sixth as a validation set.

To evaluate the performance of the model, we used the mean accuracy metric (mAP), which calculates the average of the mAP scores for the variable IoU values. It allows penalizing a large number of bounding boxes with incorrect classifications to avoid over-specialization in a few classes at the expense of weak overfitting in others.

The model was trained for 120 epochs with a learning rate of 0.01 and a momentum of 0.937. To finalize the model, 3 TTAs were applied (minor image slicing with 650×650, 750×750, 850×850, and rotation $(0°, 90°, 180°, 270°)$. To take into account the location of the image in the image (to reduce the influence of objects with larger curved features at the edge of the image), we reduced the probability by a correction factor of 0.8.

As a result of tuning the developed model, along with increasing the image set and post-processing, the accuracy of mAP object detection and recognition was improved by 0.33%, which is 81.69 compared to YOLOv11-obb.

Table 1
Dependence of mAP accuracy on changes in hyperparameters.

	Recall	mAP_{50}	mAP_{75}
$L_{ciou} + L1$	72.21	76.22	66.11
$L_{ciou} + L2$	72.34	77.51	67.31
$L_{ciou} + SmoothL1$	73.5	78.92	63.42
$L_{ciou} + SmoothL2$	75.52	81.69	69.36

4. Conclusion

To improve the efficiency and reliability of detailed interpretation of remote sensing data, we analyzed the methods of automatic image processing. As a result, the study of neural network models to solve the problem of detecting and recognizing small randomly oriented objects in satellite images revealed difficulties that reduce the accuracy of object detection and recognition.

In this study, a rotating bounding box detection model based on YOLOv11 is proposed to solve the problem of traditional horizontal detectors that have difficulty detecting targets with high density, high aspect ratio and overlapping bounding boxes. A rotation angle channel and a corresponding angular loss calculation function were added to the original YOLOv11 model. To achieve the learning effect, data label preprocessing was set up to detect and calculate the width, height, and angle of the objects. A publicly available remote sensing dataset was selected to validate the model results and assess its effectiveness. Experimental data and visual analysis showed that the YOLOv11-based model is an effective choice for detecting and recognizing small-scale multidirectional remote sensing images. Further research should focus on solving the problem of detecting and recognizing objects by detector models in adverse meteorological conditions.

Declaration on Generative AI: The authors have not employed any generative AI tools.

References

[1] S. Kovbasiuk, L. Kanevskyy, S. Chernyshuk, M. Romanchuk, Detection of vehicles on images obtained from unmanned aerial vehicles using instance segmentation, in: 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET 2020, 2020, pp. 267–271. doi:10.1109/TCSET49122.2020.235437.

[2] S. Kovbasiuk, L. Kanevskyy, M. Romanchuk, A hybrid segmentation cascade model for automatic object decoding on aerial images, Modern information technologies in the field of security and defense 35 (2019) 65–70. doi:10.33099/2311-7249/2019-35-2-65-70.

[3] D. Sudha, J. Priyadarshini, An intelligent multiple vehicle detection and tracking using modified vibe algorithm and deep learning algorithm, Soft Computing 24 (2020) 17417–17429. doi:10.1007/s00500-020-05042-z.

[4] S. A. Ahmed, D. P. Dogra, S. Kar, P. P. Roy, Unsupervised classification of erroneous video object trajectories, Soft Computing 22 (2018) 4703–4721. doi:10.1007/s00500-017-2656-x.

[5] W. Sun, D. Yan, J. Huang, C. Sun, Small-scale moving target detection in aerial image by deep inverse reinforcement learning, Soft Computing 24 (2020) 5897–5908. doi:10.1007/s00500-019-04404-6.

[6] P. Araujo, J. Fontinele, L. Oliveira, Multi-Perspective Object Detection for Remote Criminal Analysis Using Drones, IEEE Geoscience and Remote Sensing Letters 17 (2020) 1283–1286. doi:10.1109/lgrs.2019.2940546.

[7] S. Zhang, X. Mu, G. Kou, J. Zhao, Object Detection Based on Efficient Multiscale Auto-Inference in Remote Sensing Images, IEEE Geoscience and Remote Sensing Letters 18 (2021) 1650–1654. doi:10.1109/LGRS.2020.3004061.

[8] J. Qaddour, Object Detection Performance: A Comparative Study, 2023. doi:10.21203/rs.3.rs-3181849/v1.

[9] R. Girshick, Fast R-CNN, in: IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015, p. 1440–1448. doi:10.1109/ICCV.2015.169.

[10] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (2017) 1137–1149. doi:10.1109/TPAMI.2016.2577031.

[11] Y. Wu, K. Zhang, J. Wang, Y. Wang, Q. Wang, Q. Li, CDD-Net: A Context-Driven Detection Network for Multiclass Object Detection, IEEE Geoscience and Remote Sensing Letters 19 (2022) 1–5. doi:10.1109/LGRS.2020.3042465.

[12] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, X. Xue, Arbitrary-Oriented Scene Text Detection via Rotation Proposals, IEEE Transactions on Multimedia 20 (2018) 3111–3122. doi:10.1109/TMM.2018.2818020.

[13] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu, Z. Luo, R2CNN: Rotational Region CNN for Orientation Robust Scene Text Detection, CoRR abs/1706.09579 (2017). URL: http://arxiv.org/abs/1706.09579. arXiv:1706.09579.

[14] H. Law, J. Deng, CornerNet: Detecting Objects as Paired Keypoints, International Journal of Computer Vision 128 (2019) 642–656. doi:10.1007/s11263-019-01204-1.

[15] X. Zhou, D. Wang, P. Krähenbühl, Objects as Points, CoRR abs/1904.07850 (2019). URL: http://arxiv.org/abs/1904.07850. arXiv:1904.07850.

[16] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A. C. Berg, SSD: Single Shot MultiBox Detector, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, volume 9905 of *Lecture Notes in Computer Science*, Springer International Publishing, Cham, 2016, pp. 21–37. doi:10.1007/978-3-319-46448-0_2.

[17] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal Loss for Dense Object Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (2020) 318–327. doi:10.1109/TPAMI.2018.2858826.

[18] R. Khanam, M. Hussain, YOLOv11: An Overview of the Key Architectural Enhancements, CoRR abs/2410.17725 (2024). doi:10.48550/ARXIV.2410.17725. arXiv:2410.17725.

[19] W. Gai, Y. Liu, J. Zhang, G. Jing, An improved Tiny YOLOv3 for real-time object detection, Systems Science & Control Engineering 9 (2021) 314–321. doi:10.1080/21642583.2021.1901156.

[20] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, L. Zhang, DOTA: A Large-Scale Dataset for Object Detection in Aerial Images, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 3974–3983. doi:10.1109/CVPR.2018.00418.

[21] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu, Z. Luo, R2CNN: Rotational Region CNN for Orientation Robust Scene Text Detection, CoRR abs/1706.09579 (2017). URL: http://arxiv.org/abs/1706.09579. arXiv:1706.09579.

[22] S. Elfwing, E. Uchibe, K. Doya, Sigmoid-weighted linear units for neural network function approximation in reinforcement learning, Neural Networks 107 (2018) 3–11. doi:10.1016/J.NEUNET.2017.12.012.