Comparison of RetinaNet-Based Single-Target Cascading and Multi-Target Detection Models for Administrative Regions in Network Map Pictures

Kaixuan Du 1,2, Xianghong Che 2,*, Yong Wang 2, Jiping Liu 2, An Luo 2, Ruiyuan Ma 2 and Shenghua Xu 2

1 School of Resource and Environmental Science, Wuhan University, Wuhan 430079, China
2 Research Center of Geospatial Big Data Application, Chinese Academy of Surveying and Mapping, Beijing 100830, China
* Correspondence: chexh@casm.ac.cn

Abstract: There is a critical need for detection of administrative regions through network map pictures in map censorship tasks, which can be implemented by target detection technology. However, on map images there tend to be numerous administrative regions overlaying map annotations and symbols, thus making it difficult to accurately detect each region. Using a RetinaNet-based target detection model integrating ResNet50 and a feature pyramid network (FPN), this study built a multi-target model and a single-target cascading model from three single-target models by taking Taiwan, Tibet, and the Chinese mainland as target examples. Two models were evaluated both in classification and localization accuracy to investigate their administrative region detection performance. The results show that the single-target cascading model was able to detect more administrative regions, with a higher f1_score of 0.86 and mAP of 0.85 compared to the multi-target model (0.56 and 0.52, respectively). Furthermore, location box size distribution from the single-target cascading model looks more similar to that of manually annotated box sizes, which signifies that the proposed cascading model is superior to the multi-target model. This study is promising in providing support for computer map reading and intelligent map censorship.

Keywords: map pictures; RetinaNet; single-target cascading model; multi-target model; administrative regions

1. Introduction

Maps are manifestations of devising, summarizing, and modelling natural and social phenomena with a definite mathematical foundation and symbol system, and convey abundant geographic information [1]. They are of great significance in production, construction, planning, navigation, and daily travel. Furthermore, labels, indicators, and boundaries on maps represent national sovereignty and territorial integrity, thus making it crucial to produce and utilize correct maps [2]. However, there are currently numerous problematic maps on the internet, including incorrect national and regional administrative boundary depictions and missing isolated islands, which tends to result in national and regional disputes [3]. Therefore, it is imperative to quickly and accurately recognize these regions of interest on a map and then facilitate detection of incorrect regions.

The essence of recognizing regions of interest on the map is actually target detection in computer vision. Target detection aims at recognizing whether the target exists and where it is indicated by bounding boxes on the images. In the past, target detection normally relied on visual interpretation, which is inefficient and labor-intensive despite having the highest detection accuracy [4]. With the advancement of computer technology, most traditional target detection techniques developed hand-crafted (engineered) features derived from Sobel edge recognition [5], Haar-like detectors [6], histogram of oriented gradient (HOG) detectors [7], and deformable part-based models (DPMs) [8], etc., and then detect the
target by utilizing developed features and shallow classifiers or regressors such as support vector machines (SVMs) \[9\] and boosting \[10\]. Although these methods are reliable and inexpensive, feature derivation is heavily dependent on hand-engineering, thus limiting the ability to make an inference with the extracted features in real time \[11\]. It is difficult to use these methods for quickly, robustly, and automatically detecting targets.

Recently, with large amounts of data available and improved computational power, deep convolutional neural networks \[12\] and their widespread applications in image classification \[13\] have emerged, and target detection techniques based on deep learning have achieved significant progress \[14,15\]. Compared to traditional techniques, deep learning techniques can automatically extract hierarchical feature vectors/representations from the underlying data and can be disentangled through multilevel nonlinear mapping \[16\]. There are two main paradigms for target detection based on deep learning: two-stage detectors such as R-CNN \[17\] and R-FCN \[18\] and one-stage detectors such as YOLO \[19\], SSD \[20\], and RetinaNet \[21\]. The process of two-stage detection can be split into two parts: proposing regions and classifying and regressing bounding boxes. One-stage detectors directly map class predictions of different objects present at each location of the extracted feature maps without utilizing the region classification step. Two-stage detectors tend to achieve good detection performance but have decreased overall detection speed due to generating region proposals. Although one-stage detectors are sometimes not able to achieve as good a performance as the two-stage, they are comparatively more time-efficient and hence are greatly applicable for real-time target detection \[22,23\]. One-stage detectors have been largely applied under specific application scenarios, such as face detection \[24,25\], text detection \[26,27\], vehicle detection \[28\], traffic sign detection \[29\], pedestrian detection \[30,31\], and remote sensing target detection \[32,33\], but to our knowledge, they have rarely been applied to administrative region detection on a map to facilitate problematic map identification. Furthermore, it is still underexplored whether better performance could be achieved by using a deep learning model to detect multiple classes of targets simultaneously rather than only one specific class. Considering the current state-of-the-art target detector of RetinaNet by implementing a focal loss cross-entropy function and feature pyramids network (FPN) \[21,34,35\], the primary contributions of this study were to (1) deploy the RetinaNet model to detect multiple administrative regions on maps; and (2) evaluate the performance between simultaneous detection of multiple targets and cascading detection of single targets.

2. Method

2.1. RetinaNet Model Structure

The RetinaNet model is composed of two backbone networks for calculating the convolutional feature map of the input image \[21\], i.e., ResNet50 and FPN, and two task-specific subnetworks used for classification and bounding box regression (Figure 1). RetinaNet adopts ResNet50 to avoid gradient vanishing in extracting more features with the increasing convolutional layers, because it has an additional identity mapping capability. Subsequently, RetinaNet adopts the feature pyramid network (FPN) \[34\] as its backbone, which is in turn built on top of ResNet50 in a fully convolutional fashion. The fully convolutional nature enables the network to take an image of an arbitrary size and output proportionally sized feature maps at multiple levels in the feature pyramid. The FPN enhances extracted features from the convolutional network through the top-down method and horizontal connections by transforming the network from the input image to a multiscale feature pyramid, which enables the model to detect administrative regions of various sizes and increases the speed. At the same time, the area of the anchor box increases from $32 \times 32$ to $512 \times 512$, and three scales (20, 21/3, and 22/3) and three aspect ratios (1:2, 1:1, and 2:1) are used in each feature pyramid. Therefore, the feature points of each layer feature map correspond to 9 anchor boxes, which improves the average precision to a certain extent. Based on the multiscale feature pyramid, two subnetworks of classification and box regression were constructed through simple convolution operations. Specifically,
the classification subnetwork predicts the probability of each anchor box class, which is a fully convolutional network (FCN) attached to each FPN level and shares parameters. The subnet consists of four $3 \times 3$ convolutional layers with 256 filters, followed by ReLU activations. Then, another $3 \times 3$ convolutional layer with $K \times A$ filters is connected, where $K$ is the number of categories of detection targets, and $A$ is the number of anchor boxes. The category is predicted at each position through sigmoid activation (Figure 1). Similarly, the regression subnet is attached to each feature map of the FPN in parallel to the classification subnet. The design of the regression subnet is identical to that of the classification subnet, except that the last convolutional layer is $3 \times 3$ with $4 \times A$ filters (Figure 1).

Figure 1. Basic network model structure of RetinaNet.

2.2. Focal Loss

RetinaNet introduces a new loss function, focal loss (FL). Focal loss is an improved CE (cross-entropy) loss function. The model with focal loss reduces the contribution of negative samples in model training by passing a weakening index on the basis of the original CE loss. Consequently, it solves the problem of unbalanced positive and negative samples in the training of the target detection model and improves model accuracy [21]. The loss in RetinaNet is a multi-task loss of classification (hereafter namely $L_{\text{cls}}$) and regression (hereafter namely $L_{\text{reg}}$), which can be written as:

$$L = \lambda L_{\text{reg}} + L_{\text{cls}}, \quad (1)$$

In Equation (1), $\lambda$ is a hyperparameter that controls the balance between the two task losses. The classification loss $L_{\text{cls}}$ for each anchor is expressed as follows:

$$L_{\text{cls}} = - \sum_{i=1}^{K} (y_i \log(p_i)(1 - p_i)^\gamma a_i + (1 - y_i) \log(1 - p_i) p_i^\gamma (1 - a_i)), \quad (2)$$

where $K$ denotes the number of classes; $y_i$ equals 1 if the groundtruth belongs to the $i$-th class and 0 otherwise; $p_i$ is the predicted probability for the $i$-th class; $\gamma \in (0, +\infty)$ is a focusing parameter; and $a_i \in [0, 1]$ is a weighting parameter for the $i$-th class. In reality, negative samples that are matched to the target in an image account for a large portion of inputs, which can be easily classified by the detector as background and contribute no useful learning signal, overwhelming the loss and computed gradients and leading to degenerated models. The focal loss function introduces the focusing parameter $\gamma$ to downweight the loss assigned to easily classified examples. This effect increases as the value of $\gamma$ increases and makes the network focus more on positive samples. The balancing parameter $\alpha$ is also useful for addressing class imbalance.

The regression loss $L_{\text{reg}}$ represents smooth $L_1$ loss of the bounding box regression [21], and can be defined as:

$$L_{\text{reg}} = \sum_{j \in \{x,y,w,h\}} \text{smooth}_{L_1}(p_i - T_i^j), \quad (3)$$
Here, for each anchor A with a match, the regression subnet predicts four numbers as $P_i = (P_{ix}, P_{iy}, P_{iw}, P_{ih})$, which represent the center coordinates, width, and height of the anchor bounding box. A regression target $T_i$ is computed as the offset between the anchor A and the groundtruth G as follows:

$$T_{ix}^i = \frac{(G_{ix}^i - A_{ix}^i)}{A_{iw}^i},$$

$$T_{iy}^i = \frac{(G_{iy}^i - A_{iy}^i)}{A_{ih}^i},$$

$$T_{iw}^i = \log \left( \frac{G_{iw}^i}{A_{iw}^i} \right),$$

$$T_{ih}^i = \log \left( \frac{G_{ih}^i}{A_{ih}^i} \right),$$

where smooth$_{L1}(x)$ is smooth L1 loss, which can be defined as:

$$\text{Smooth}_{L1}(x) = \begin{cases} 0.5x^2 & |x| < 1 \\ |x| - 0.5 & |x| \geq 1 \end{cases}.$$  

2.3. Single-Target Cascading and Multi-Target Detection Using RetinaNet Models

Considering the different sizes and shapes of administrative regions on the maps, this study designed two strategies using the RetinaNet architecture (hereafter referred to as single-target cascading detection and multi-target detection) to explore their performance for detecting administrative regions on the maps. Specifically, in the single-target cascading detection model, the RetinaNet model was trained for each class of targets separately. That is to say, each class of targets had a specific model. For detection, a map goes through these models one by one, and the detected results are aggregated by the union operator to derive the final detection result. The principle of target extraction by the single-target cascading detection model is shown in Equation (9).

$$F_M(x) = \bigcup_{m=1}^{n} h(x; a_m),$$

where $h(x; a_m)$ is a single-target detection model, $a_m$ is parameters in the model, and $x$ is the input image. $F_M(x; P)$ is the final target detection model, and $n$ is the number of target classes.

In contrast, the multi-target detection strategy takes all classes of targets as the input for one RetinaNet model and generates a trained model for simultaneously detecting multiple class targets. The specific structure of the two strategies is shown in Figure 2.
3. Experiments

In this section, we introduce the dataset and the evaluation indexes used in our experiment and then illustrate the detailed information of the experiment.

3.1. Dataset

Three administrative regions (targets) of the map are used in this paper, which include Taiwan province, Tibet province, and the Chinese mainland in China. Using an orientation crawler approach, 3991 map images were derived from an internet image search engine using the keywords Taiwan, Tibet, Chinese mainland, and map. We randomly divided 3192 images into a training set and 799 images into a test set. Map images from the internet have a broad range of image size. Both large and small target regions were included in the sample dataset. The size of Taiwan varied from 24 pixels $\times$ 14 pixels to 3240 pixels $\times$ 1819 pixels. The size of Tibet varied from 24 pixels $\times$ 14 pixels to 3240 pixels $\times$ 1819 pixels, and the Chinese mainland varied from 32 pixels $\times$ 21 pixels to 2496 pixels $\times$ 2863 pixels. We split samples with different sizes into the training and test datasets randomly. Furthermore, the datasets in this study contained different kinds of thematic maps, such as seismic zone maps, climate maps, transportation maps, and terrain mountain maps, etc., as shown in Figure 3.

The test set was intended to provide an unbiased evaluation of a trained model using the training set. The specific distribution is shown in Table 1. The three targets in the map images were manually annotated using a web-based image annotation tool. The tool outputs an annotation file with an interactive drawing of a bounding box containing all the pixels of the target, which include the directory of each image, the coordinates of the top left corner for the annotated bounding box, the width and height of the annotated bounding box, and the name of the target (Table 2). The principle of manual annotation is to use the smallest possible box to completely cover the targets but get rid of the useless background.

Table 1. Sample distribution of different targets.

| Region of Interest | Training Dataset | Test Dataset | Total |
|--------------------|------------------|--------------|-------|
| Target 1           | Taiwan           | 2151         | 538   | 2689  |
| Target 2           | Tibet            | 582          | 146   | 728   |
| Target 3           | Chinese mainland | 459          | 115   | 574   |
| Total              |                  | 3192         | 799   | 3991  |
Figure 3. Different thematic maps: (a) seismic zone; (b) climate; (c) transportation and mountain terrain.

Table 2. Target annotation format.

| No. | path_img_file       | box_x  | box_y  | Width | Height | Label         |
|-----|---------------------|--------|--------|-------|--------|---------------|
| 1   | dataset/image_0001.jpg | 890    | 659    | 944   | 743    | Taiwan        |
| 2   | dataset/image_0002.jpg | 775    | 631    | 845   | 721    | Taiwan        |
| 3   | dataset/image_0003.jpg | 36     | 57     | 762   | 535    | Xizang        |
| 4   | dataset/image_0004.jpg | 5      | 51     | 536   | 316    | Xizang        |
| 5   | dataset/image_0005.jpg | 5      | 2      | 341   | 289    | Chinese mainland |
| 6   | dataset/image_0006.jpg | 95     | 93     | 666   | 546    | Chinese mainland |

3.2. Evaluation Metrics

We used four evaluation metrics, including intersection over union (IOU), precision, recall, and harmonic mean of precision and recall [36,37]. IOU is used to measure how much our predicted boundary overlapped with the ground truth (the target’s real boundary), which calculated the coincidence degree between the predicted box and the ground truth box. IOU is defined by Equation (10), where \( B_p \) represents the predicted bounding box and \( B_{gt} \) represents the ground truth bounding box. The threshold of IOU indicates whether the detection is valid or not.

\[
IOU = \frac{\text{area}(B_{gt} \cap B_p)}{\text{area}(B_{gt} \cup B_p)},
\]  

(10)

Here, if the predicted bounding box overlapped with the annotated bounding box and exceeded the IOU threshold (i.e., 0.5 in classifying whether the prediction was a true positive or a false positive), the predicted bounding box represented the administrative region sample; otherwise, it was the background sample.

Precision and recall represent the classification accuracy of the model, where precision measures the accuracy of detected targets and recall measures the integrity of detection. An f1 score is a comprehensive evaluation metric that measures the accuracy of a classification
model by calculating the average of precision and recall. The three metrics can be calculated as follows:

\[
\text{precision} = \frac{TP}{TP + FP},
\]

\[
\text{recall} = \frac{TP}{TP + FN},
\]

\[
\text{f}_1\text{score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}},
\]

where \(TP\) denotes the number of correctly predicted administrative region samples; \(FN\) denotes the number of wrongly classified samples; and \(FP\) denotes the number of incorrectly identified samples.

Average precision (AP) balances the precision (P) and recall (R) values, reflecting the performance of the model in each class, which is the area under the precision-recall curve, as shown in Equation (14) [38]. The \(mAP\) is the average of \(AP\) for target classes, which is used to show the model’s advantages and disadvantages across all classes (Equation (15)). This study used the value of \(mAP\) when the IOU threshold = 0.5 [38,39].

\[
AP(n) = \int_0^1 p(r_n)dr_n,
\]

\[
mAP = \frac{1}{n} \sum_{i=1}^{n} AP(n),
\]

3.3. Implementation Details

Model training was carried out on the Ubuntu18.04.6 LTS operating system. The computer hardware configuration was as follows: Intel (R) Core (TM) i9-10900X, 64 G, NVIDIA GeForce RTX 3090, and GPU with 24 G memory. The model training environment was Python 3.6 and Keras 2.4.3. The ResNet50 backbone network was initialized with ImageNet training set parameters. The initial learning rate was 0.0001 during the training process. A total of 100 epochs were trained, and the batch was set to one image. Weight attenuation was set to 0.0001 and momentum was set to 0.9. \(\gamma\) was set to 2, and \(\alpha\) was set to 0.25 in the focal loss function.

3.4. Experimental Results

Figure 4a–c shows the total loss, classification loss, and regression loss curves of three training models for three targets, respectively, and Figure 4d shows the training model with all three targets. It can be seen that the loss curves of both the single-target and multi-target detection models flatten out, and the models tend to converge after 100 epochs of training processing. However, both the classification loss and regression loss of the single-target model is lower than the multiple target training model, which demonstrates the single-target training model performs better. In addition, the classification losses are always lower than the regression losses for all training models. Although the training loss of the multi-target detection model are higher than the single-target model, the multi-target detection model needed fewer epochs—about 40—to get a relatively stable regression loss value. The training duration of three single-target detection models was 24,546 s, 7303 s, and 5211 s, respectively. The training duration of the multi-target detection model was 32,692 s.

The single-target and multi-target trained detection models were also evaluated with the testing samples, as shown in Tables 3 and 4, respectively. The detection results from single-target cascading detection models have higher \(P\), \(R\), and \(f_1\text{score}\) values (0.80, 0.93 and 0.86, respectively) than the multi-target model (0.77, 0.44, and 0.56, respectively), which indicates that the single-target cascading model is superior to the multi-target model. Moreover, the multi-target model has apparent omission errors, with an \(R\) of 0.44, especially for the Chinese mainland (0.05).
Figure 4. Training loss of single-target and multi-target detection models: (a) Training loss of model for target 1 (Taiwan); (b) Training loss of model for target 2 (Tibet); (c) Training loss of model for target 3 (Chinese mainland); (d) Training loss of model for all three targets.

Table 3. Accuracy statistics of different targets with the single-target cascading detection model.

|                  | Single-Target Model | Single-Target Cascading Detection Models |
|------------------|---------------------|-----------------------------------------|
|                  | Taiwan   | Tibet      | Chinese Mainland | Taiwan, Tibet, and Chinese Mainland |
| Precision (P)    | 0.92     | 0.77       | 0.52             | 0.80                                 |
| Recall (R)       | 0.91     | 0.96       | 0.94             | 0.93                                 |
| f1_score         | 0.92     | 0.86       | 0.67             | 0.86                                 |

Table 4. Accuracy statistics of different targets with the multi-target detection model.

|                  | Multi-Target Model |
|------------------|--------------------|
|                  | Taiwan | Tibet  | Chinese Mainland | Taiwan, Tibet, and Chinese Mainland |
| Precision (P)    | 0.94   | 0.53   | 0.3              | 0.77                                 |
| Recall (R)       | 0.78   | 0.31   | 0.05             | 0.44                                 |
| f1_score         | 0.85   | 0.39   | 0.10             | 0.56                                 |

The results of three administrative regions detected by single-target cascading and multi-target models were visually compared with the manual annotations (ground truth), and some examples are shown in Figure 5. The rectangular frames are manually annotated, while the frames in the last two rows represent their prediction results from the single-target cascading and multi-target models, respectively. The recognized regions are marked by different rectangular boxes, highlighting the different administrative regions in the maps. It is clear that the single-target cascading model correctly detected three administrative regions while the multi-target detection model only detected the Taiwan target. A large region such as the Chinese mainland was difficult for the multi-target detection model to detect under the influence of a more complicated background compared to the other two targets, which is consistent with the quantitative results of Tables 3 and 4.
Figure 5. Annotations of different targets in maps (top row) and detected results from the single-target cascading model (middle row) and multi-target model (bottom row). (Different colors of the boxes indicate different types of targets, and labels indicate the target types; score is for classification confidence; IOU is for localization confidence).

Figure 6 shows the P–R curves of the Taiwan, Tibet, and Chinese mainland regions detected by the single-target cascading and multi-target models. Obviously, Taiwan has the highest precision for two categories of model among three administrative regions. The single-target cascading model was able to maintain a higher precision when having the same recall compared to the multi-target model, especially for Tibet and the Chinese mainland. The precision of the multi-target mode rapidly decreased with the increasing recall, but the single-target cascading model could better balance precision and recall.

The AP values for each administrative region are also compared in Figure 7. All AP values of the three administrative regions from the single-target cascading model were higher than the multi-target model, especially for Tibet. The mean AP (mAP) is 0.85 and 0.52 for the single-target cascading and multi-target detection models, respectively. This also implies the superiority of the single-target cascading detection model in detecting regions of interest on maps.

As shown in Figure 8, the ground truth and predicted sizes from the single-target cascading model have more similar distributions than the multi-target model. Box size was primarily distributed in the range of (0, 10,000), with the largest number in the range of (500, 1000). This comparison demonstrates the single-target cascading model had more precise location regression results.
4. Discussion

With the advances in computer vision technologies and the overwhelming availability of open-source big data, deep learning-based target detection has become a popular research topic. However, previous studies primarily focused on entity targets such as faces, pedestrians, vehicles, traffic signs, text, roads, etc. [24–33]. These targets normally have specific colors, textures, or shapes. More importantly, these targets tend to be visually obvious and uncovered by other elements, which enable a deep learning-based model to easily extract features for detection. In contrast, unlike traditional maps created by professional cartographers in the past, current map resources vary greatly because of free creation, publishing, editing, and sharing on the Internet. This inevitably leads to different
cartographic semantics, data descriptions, drawing standards, and design patterns on the administrative regions of maps as well as many covered elements on the administrative regions, which complicate the desired region detection [35,40–43]. Although there exist some studies to detect cartographic semantics using deep learning techniques, most of them focused on detecting map text and symbols using convolutional neural networks (CNNs) and generative adversarial networks (GANs) [44–49]. It has rarely been explored how to intelligently detect which administrative regions are included in a map. This study built region-detection models with high accuracy to detect the Chinese mainland, Taiwan, and Tibet, which is capable of migration, and can be extended to any administrative region.

This study proposed a RetinaNet-based cascading model to detect Chinese mainland, Taiwan, and Tibet maps and presented a series of experiments to evaluate its performance. The results show that the model obtains higher mAP for three regions in the map. We used the loss function and FPN to strengthen the learning of administrative region features, which further advanced the utilization rate of features of regions of interest and inhibited feature learning on the background or adjacent regions to improve the detection effect for regions of interest and to accelerate the convergence speed of the final results. From the test results of this model, it can be concluded that this model has a good auxiliary effect on administrative region screening and has certain application prospects, such as intelligent recognition of problematic maps and map-to-text conversion.

Meanwhile, we evaluated the performance of single-target cascading and multi-target detection with RetinaNet-based models, which indicated the superiority of single-target cascading detection. Multi-target training is a more complex learning process than the single-target one. That is to say, the feature learning for multiple targets requires more parameters to train the model. Using the same training data, it is easier to achieve a single-target detection. The process of cascaded single-target models for detection is similar to the process of constructing a piecewise function where AdaBoost theory is employed to combine multiple classifiers to obtain a robust classifier [50,51].

There are also some limitations for the proposed RetinaNet-based single-target cascading model. For example, the Chinese mainland region had the lowest accuracy, and this is mainly because the Chinese mainland region has a more complex background compared to the Taiwan and Tibet regions. FPN was adopted to increase the richness of features represented by each pyramid ladder in the pyramid-shaped model, using a multiscale feature extracted from the ResNet50 layer, but the omission error occurred for Chinese mainland region. Some improved strategies, such as adding a multiple-level feature graph [34], adopting adaptive training sample selection [49], and designing a loss weight adjustment [52] can be investigated to enhance the target training for the proposed model in the future.

5. Conclusions

In this paper, taking Taiwan, Tibet, and the Chinese mainland as target examples, a single-target cascading model was constructed by training three single-target models based on the widely-used RetinaNet model. At the same time, a multi-target model was built with all three target samples. Their comparisons for classification and location evaluation shows the RetinaNet-based single-target cascading model can better detect administrative regions on the network map pictures. This model also can be extended to any other administrative region. This study will greatly reduce the workload for map review specialists and improve their work efficiency.

Author Contributions: Conceptualization, K.D. and Y.W.; methodology, K.D. and Y.W.; software, K.D.; validation, K.D. and X.C.; formal analysis, X.C.; investigation, X.C.; resources, Y.W.; data curation, K.D. and R.M.; writing—original draft preparation, X.C.; writing—review and editing, K.D., X.C., J.L., and A.L.; visualization, K.D.; supervision, Y.W.; project administration, X.C. and Y.W.; funding acquisition, X.C. and S.X. All authors have read and agreed to the published version of the manuscript.
Funding: This research was funded by National Nature Science Foundation of China, grant numbers 41901379 and 42071384, and the Basic Scientific Research Foundation of the Chinese Academy of Surveying and Mapping, grant number AR2205.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Jiayao, W.; Yi, C. Discussions on the Attributes of Cartography and the Value of Map. Acta Geod. Cartogr. Sin. 2015, 44, 237.
2. Huang, L. Research on the method of fast mining internet problem map picture. Geomat. Spat. Inf. Technol. 2017, 40, 92–96.
3. Zhou, J.; Wu, X. Problem Map Picture current Situation Analysis and Countermeasures. Geomat. Technol. Equip. 2018, 20, 41–43.
4. Ren, J.; Liu, W.; Li, Z.; Li, R.; Zhai, X. Intelligent Detection of “Problematic Map” Using Convolutional Neural Network. Geomat. Inf. Sci. Wuhan Univ. 2021, 46, 570–577.
5. Roberts, L.G. Machine Perception of Three-Dimensional Solids. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1963.
6. Lienhart, R.; Maydt, J. An Extended Set of Haar-like Features for Rapid Object Detection. In Proceedings of the International Conference on Image Processing, Rochester, NY, USA, 22–25 September 2002; Volume 1, pp. 900–903.
7. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
8. Felzenszwalb, P.; McAllester, D.; Ramanan, D. A Discriminatively Trained, Multiscale, Deformable Part Model. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
9. Hsu, C.-W.; Lin, C.-J. A Comparison of Methods for Multiclass Support Vector Machines. IEEE Trans. Neural Netw. Neurocomputing 2002, 13, 415–425.
10. Schneiderman, H.; Kanade, T. Object Detection Using the Statistics of Parts. Int. J. Comput. Vis. 2004, 56, 151–177. [CrossRef]
11. Cho, S.M.; Kim, Y.-G.; Jeong, J.; Kim, I.; Lee, H.; Kim, N. Automatic Tip Detection of Surgical Instruments in Biportal Endoscopic Spine Surgery. Comput. Biol. Med. 2021, 133, 104384. [CrossRef]
12. Cao, G.; Xie, X.; Yang, W.; Liao, Q.; Shi, G.; Wu, J. Feature-Fused SSD: Fast Detection for Small Objects. In Proceedings of the Ninth International Conference on Graphic and Image Processing (ICGIP 2017), Qingdao, China, 10 April 2018; Volume 10615, pp. 381–388.
13. Tripathi, S.; Dane, G.; Kang, B.; Bhaskaran, V.; Nguyen, T. Lcdet: Low-Complexity Fully-Convolutional Neural Networks for Object Detection in Embedded Systems. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops 2017, 411–420. [CrossRef]
14. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected Crfs. arXiv 2014, arXiv:1412.7062.
15. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-Cnn: Towards Real-Time Object Detection with Region Proposal Networks. Adv. Neural Inf. Process. Syst. 2017, 39, 1137–1149. [CrossRef] [PubMed]
16. Sharma, V.; Mir, R.N. A Comprehensive and Systematic Look up into Deep Learning Based Object Detection Techniques: A Review. Comput. Sci. Rev. 2020, 38, 100301. [CrossRef]
17. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [CrossRef]
18. Dai, J.; Li, Y.; He, K.; Sun, J. R-Fcn: Object Detection via Region-Based Fully Convolutional Networks. Adv. Neural Inf. Process. Syst. 2016, 29, 379–387.
19. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
20. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single Shot Multibox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37. [CrossRef]
21. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. IEEE Trans. Pattern Anal. Mach. Intell. 2020, 42, 318–327. [CrossRef]
22. Jiao, L.; Zhang, F.; Liu, F.; Yang, S.; Li, L.; Feng, Z.; Qu, R. A Survey of Deep Learning-Based Object Detection. IEEE Access 2019, 7, 128837–128868. [CrossRef]
23. Wu, X.; Sahoo, D.; Hoi, S.C.H. Recent Advances in Deep Learning for Object Detection. Neurocomputing 2020, 396, 39–64. [CrossRef]
24. Zafeiriou, S.; Zhang, C.; Zhang, Z. A Survey on Face Detection in the Wild: Past, Present and Future. Comput. Vis. Image Underst. 2015, 138, 1–24. [CrossRef]
25. Wang, N.; Gao, X.; Tao, D.; Yang, H.; Li, X. Facial Feature Point Detection: A Comprehensive Survey. Neurocomputing 2018, 275, 50–65. [CrossRef]
26. Ye, Q.; Doermann, D. Text Detection and Recognition in Imagery: A Survey. IEEE Trans. Pattern Anal. Mach. Intell. 2014, 37, 1480–1500. [CrossRef]

27. Yin, X.-C.; Zuo, Z.-Y.; Tian, S.; Liu, C.-L. Text Detection, Tracking and Recognition in Video: A Comprehensive Survey. IEEE Trans. Image Process. 2016, 25, 2752–2773. [CrossRef] [PubMed]

28. Sivaraman, S.; Trivedi, M.M. Looking at Vehicles on the Road: A Survey of Vision-Based Vehicle Detection, Tracking, and Behavior Analysis. IEEE Trans. Intell. Transp. Syst. 2013, 14, 1773–1795. [CrossRef]

29. Mogelmoser, A.; Trivedi, M.M.; Moeslund, T.B. Vision-Based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey. IEEE Trans. Intell. Transp. Syst. 2012, 13, 1484–1497. [CrossRef]

30. Geronimo, D.; Lopez, A.M.; Sappa, A.D.; Graf, T. Survey of Pedestrian Detection for Advanced Driver Assistance Systems. IEEE Trans. Pattern Anal. Mach. Intell. 2010, 32, 1239–1258. [CrossRef] [PubMed]

31. Brunetti, A.; Buongiorno, D.; Trotta, G.F.; Bevilacqua, V. Computer Vision and Deep Learning Techniques for Pedestrian Detection and Tracking: A Survey. Neurocomputing 2018, 300, 17–33. [CrossRef]

32. Li, W.; Du, Q. A Survey on Representation-Based Classification and Detection in Hyperspectral Remote Sensing Imagery. Pattern Recognit. Lett. 2016, 83, 115–123. [CrossRef]

33. Cheng, G.; Han, J. A Survey on Object Detection in Optical Remote Sensing Images. ISPRS J. Photogramm. Remote Sens. 2016, 117, 11–28. [CrossRef]

34. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 936–944. [CrossRef]

35. Li, H.; Liu, J.; Zhou, X. Intelligent Map Reader: A Framework for Topographic Map Understanding with Deep Learning and Gazetteer. IEEE Access 2018, 6, 25363–25376. [CrossRef]

36. Xiong, X.; Duan, L.; Liu, L.; Tu, H.; Yang, P.; Wu, D.; Chen, G.; Xiong, L.; Yang, W.; Liu, Q. Panicle-SEG: A Robust Image Segmentation Method for Rice Panicles in the Field Based on Deep Learning and Superpixel Optimization. Plant Methods 2017, 13, 1–15. [CrossRef]

37. Ma, B.; Liu, Z.; Jiang, F.; Yan, Y.; Yuan, J.; Bu, S. Vehicle Detection in Aerial Images Using Rotation-Invariant Cascaded Forest. IEEE Access 2019, 7, 59613–59623. [CrossRef]

38. Liu, S.; Cai, T.; Tang, X.; Zhang, Y.; Wang, C. Visual Recognition of Traffic Signs in Natural Scenes Based on Improved RetinaNet. Entropy 2022, 24, 112. [CrossRef]

39. Chen, J.; Li, P.; Xu, T.; Xue, H.; Wang, X.; Li, Y.; Lin, H.; Liu, P.; Dong, B.; Sun, P. Detection of Cervical Lesions in Colposcopic Images Based on the RetinaNet Method. Biomed. Signal Process. Control 2022, 75, 103589. [CrossRef]

40. Jaeger, P.F.; Kohl, S.A.A.; Bickelhaupt, S.; Isensee, F.; Kuder, T.A.; Schlemmer, H.-P.; Maier-Hein, K.H. Retina U-Net: Embracingly Simple Exploitation of Segmentation Supervision for Medical Object Detection. Proc. Mach. Learn. Health Workshop PMLR 2020, 116, 171–183.

41. Chiang, Y.-Y.; Leyk, S.; Nazari, N.H.; Moghaddam, S.; Tan, T.X. Assessing the Impact of Graphical Quality on Automatic Text Recognition in Digital Maps. Comput. Geosci. 2016, 93, 21–35. [CrossRef]

42. Miao, Q.; Liu, T.; Song, J.; Gong, M.; Yang, Y. Guided Superpixel Method for Topographic Map Processing. IEEE Trans. Geosci. Remote Sens. 2016, 54, 6265–6279. [CrossRef]

43. Liu, T.; Xu, P.; Zhang, S. A Review of Recent Advances in Scanned Topographic Map Processing. Neurocomputing 2019, 328, 75–87. [CrossRef]

44. Pezeshk, A.; Tutwiler, R.L. Automatic Feature Extraction and Text Recognition from Scanned Topographic Maps. IEEE Trans. Geosci. Remote Sens. 2011, 49, 5047–5063. [CrossRef]

45. Zhou, X. GeoAI-Enhanced Techniques to Support Geographical Knowledge Discovery from Big Geospatial Data. Ph.D. Thesis, Arizona State University, Tempe, AZ, USA, 2019.

46. Zhou, X.; Li, W.; Arundel, S.T.; Liu, J. Deep Convolutional Neural Networks for Map-Type Classification. arXiv 2018, 1805.10402.

47. Uhl, J.H.; Leyk, S.; Chiang, Y.Y.; Duan, W.; Knoblock, C.A. Undefined Automated Extraction of Human Settlement Patterns from Historical Topographic Map Series Using Weakly Supervised Convolutional Neural Networks. IEEE Access 2019, 8, 6978–6996. [CrossRef]

48. Courtila, A.; El Ayedi, A.; Touya, G.; Zhang, X. Exploring the Potential of Deep Learning Segmentation for Mountain Roads Generalisation. ISPRS Int. J. Geo-Inf. 2020, 9, 338. [CrossRef]

49. Pang, J.; Chen, K.; Shi, J.; Feng, H.; Ouyang, W.; Lin, D. Libra R-Cnn: Towards Balanced Learning for Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 821–830. [CrossRef]

50. An, T.-K.; Kim, M.-H. A New Diverse AdaBoost Classifier. In Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence, Sanya, China, 23–24 October 2010; Volume 1, pp. 359–363.

51. Pan, S.; Wang, Y.; Liu, C.; Ding, X. A Discriminative Cascade CNN Model for Offline Handwritten Digit Recognition. In Proceedings of the 2015 14th IAPR International Conference on Machine Vision Applications (MVA), Tokyo, Japan, 18–22 May 2015; pp. 501–504.

52. Luo, Y.; Jiang, P.; Duan, C.; Zhou, B. Small Object Detection Oriented Improved-RetinaNet Model and Its Application. Comput. Sci. 2021, 48, 233–238.