Hindawi
Scientific Programming
Volume 2022, Article ID 4223295, 10 pages
https://doi.org/10.1155/2022/4223295

Research Article

Animation Character Detection Algorithm Based on Clustering and Cascaded SSD

Yuan Wang

School of Design and Art, Xijing University, Xi'an, Shaanxi 710123, China

Correspondence should be addressed to Yuan Wang; 20070045@xijing.edu.cn

Received 25 November 2021; Accepted 11 December 2021; Published 7 January 2022

Academic Editor: Le Sun

With the evolution of the Internet and information technology, the era of big data is a new digital one. Accordingly, animation IP has been more and more widely welcomed and concerned with the continuous development of the domestic and international animation industry. Hence, animation video analysis will be a good landing application for computers. This paper proposes an algorithm based on clustering and cascaded SSD for object detection of animation characters in the big data environment. In the training process, the improved classification Loss function based on Focal Loss and Truncated Gradient was used to enhance the initial detection effect. In the detection phase, this algorithm designs a small target enhanced detection module cascaded with an SSD network. In this way, the high-level features corresponding to the small target region can be extracted separately to detect small targets, which can effectively enhance the detection effect of small targets. In order to further improve the effect of small target detection, the regional candidate box is reconstructed by a *k*-means clustering algorithm to improve the detection accuracy of the algorithm. Experimental results demonstrate that this method can effectively detect animation characters, and performance indicators are better than other existing algorithms.

1. Introduction

In recent years, IP (intellectual property) has ushered in the outbreak of various forms of industry, such as literature, music, games, animation, and film and television industries. In addition, the development of related industries with IP as the core also promotes the cross-border integration of different industries. All kinds of enterprises in the whole industrial chain have started cross-industry and cross-industry new exploration in different forms. Various IP resources are utilized to promote operation and development [1]. Big data has strong functionality in the era of network science and technology with highly developed information sharing. However, the transformation of big data as new thinking will be missed out if we only see the functionality of big data. Therefore, animation circles can apply the function and thinking of big data to promote the development of animation IP [2].

Among all kinds of IP, animation IP has received more and more attention with the continuous development of the domestic and international animation industry, such as the birth of "Nezha's Devil boy" and "the big fish Begonia" in China, and "the pirate king" and "spider man" in foreign countries. In addition to creating high-quality works in this field, such excellent animation IP often extends to a variety of surrounding industries. Borrowing the mature IP of the industry, it has made more profits in the pan entertainment field and expanded its influence [3]. Therefore, the intelligent detection and recognition of animation characters and their surroundings (such as Cosplay) can help cultivate users' interest in animation characters, encourage users to consume products around animation characters, and realize pan entertainment of animation IP.

Object detection is one of the basic tasks of computer vision, which is widely used in the fields of unmanned driving, safety systems, and defect detection. Target detection technology is mainly divided into three research directions. The first target detection direction is the traditional target detection method. In this method, feature descriptors are constructed to extract features, and then classifiers are used to classify features to achieve target detection [4]. Typical representatives are HOG, LBP, and Haar.

The second target detection direction is a two-step target detection algorithm, which first recommends regions and then classifies targets. Typical representatives are regions with volatile neural network features (R-CNN) and fast R-CNN. R-CNN framework and Fast R-CNN framework were proposed by Girshick to improve the accuracy of target detection [5, 6]. After that, Ren et al. proposed a Faster R-CNN network, in which the candidate regions were proposed by RPN (region proposal network). In the final feature map, the objects in the original image are much smaller and difficult to locate. Therefore, Faster R-CNN cannot solve the problem of small object detection [7]. At the same time, a complete convolutional network (FCN) has been proposed and proved to be good at semantic segmentation tasks. FCN combines convolution and pooling networks to receive image and output feature maps. Feature map uses deconvolution layer to obtain output image with the same size as input image [8].

The third target detection direction is end-to-end target detection, which uses a deep learning network for one-step detection. Its typical representatives are You Only Look Once (YOLO), SSD, etc. Redmon has proposed the YOLO algorithm, which is an end-to-end network architecture. The input of the network is image content, and the output is the information of boundary box and related class probability. Yolov3 [9], the third version of Yolo, connects the high-level network with the low-level network to obtain more meaningful semantic information from the fine-grained information in the high-level function and early function diagrams [10].

In the end-to-end single-stage target detection algorithm, SSD (single shot multibox detector) uses a layered detection method. SSD has good detection speed and accuracy, being one of the best target detection algorithms in industrial production. SSD algorithm takes into account the detection speed of the YOLO algorithm and the detection accuracy of the fast RCNN algorithm [11]. However, the effect of small target detection is general [12] since SSD uses Conv4_ 3 when a low-level feature is applied to small target detection. Besides, the number of low-level feature convolution layers is small, and there is the problem of insufficient feature extraction. In response to this problem, researchers have made many improvements to the SSD algorithm. For example, DSSD adds context information to the algorithm. The improved algorithm uses the deconvolution operation to a one-way fusion of high and low-level features, which improves the overall target detection effect and also improves the detection of small targets [13]. RSSD integrates the characteristics of different layers through rainbow concatenation. While increasing the feature map relationship between different layers, it also increases the number of feature maps in different layers, which improves the overall target detection effect and also improves the detection of small targets [14].

Although the above algorithm has achieved good detection results, there are still some shortcomings. The original SSD algorithm is not good for small target detection, which is mainly caused by three reasons. The first reason is that the training samples are unbalanced. The second reason is that the underlying feature representation ability is weak, and it is difficult to accurately classify small targets. The third reason is that the feature of a small target area is very small, which cannot be detected in a high-level feature detector. However, DSSD and RSSD adapt feature fusion, resulting in large network parameters and more calculation, which makes the detection speed very slow. In order to solve the problems in the above methods, this paper proposes a target detection algorithm based on clustering and cascaded SSD. The advantages of the algorithm are as follows:

(1) To solve the problem of unbalanced training samples, the proposed method designed an improved loss function based on Focal Loss and Truncated gradient (TG). It makes the training more focused on the samples with large loss values that are not easy to classify. The trained model can better detect the target.

(2) Aiming at the problem that the underlying feature representation ability is not enough and cannot be classified accurately, a small target enhancement detection module is designed to cascade with the SSD network. The module detects small targets in high-level features with sufficient position information and representation ability.

(3) To solve the problem that the feature of a small target area is too small to be detected in the high-level feature detector, the upsampling method of the bicubic interpolation algorithm is adopted. It samples the bottom feature of the small target area to the size of the original bottom feature so that the small target can be detected in the top feature detector.

(4) For real-time detection needs fast detection speed, this paper uses phased detection to detect the target, making the computation of the network not increase much. And the real-time performance is good.

(5) In order to solve the problem of low accuracy when selecting the default box, the k-means clustering algorithm is used to reconstruct the regional candidate box to improve the detection accuracy of the algorithm.

2. Related Work

2.1. Classic SSD Algorithm. SSD algorithm, proposed by Wei Liu, is one stage class algorithm. The algorithm is characterized by fast detection speed and good detection accuracy, which can meet the general industrial needs. It is one of the most widely used detection algorithms. The SSD algorithm combines the regression ideas in the YOLO algorithm and the Anchor mechanism in the Faster-RCNN algorithm. It uses multiscale regions at various locations in the entire image for regression, which not only maintains the fast characteristics of the YOLO algorithm but also ensures that the window prediction is as accurate as the Faster-RCNN algorithm. The core of the SSD algorithm is to use convolution kernels on feature maps of different scales to predict the category and coordinate offset of a series of Default Bounding Boxes.

The network adopts a layered detection mode. A low level detects a small target, and a high level detects a big target. Finally, the redundant boundary boxes are removed through NMS (nonmaximum suppression) to obtain the final detection result. Since the semantic information of a small target is not enough at the bottom level, and the location information at the high level is not enough as well, the SSD has a good effect on big target detection. However, the detection effect of small targets is relatively poor, and there are some deficiencies in the target detection task with more small targets.

The algorithm in this paper uses layered detection mode for reference. The detection is divided into two stages. In the first stage, the trained model is used to extract the high-level features of the image and detect the target, respectively, to obtain the small target area with an unknown specific category and the big target border with a known specific category. In the second stage, the small target enhancement detection module is cascaded with the SSD network. The high-level features corresponding to the small target area are extracted separately to detect the small target. This kind of detection adopts the phased form, which can improve the detection effect and has good real-time performance.

2.2. Focal Loss. Focal Loss is a new loss function for target detection proposed by Kaiming and RB. The loss function is mainly used to deal with the imbalance of training samples. By reducing the weight of easy-to-classify samples, Focal Loss makes the model pay more attention to hard-to-classify samples in training. In this way, the training effect is enhanced and the detection result is improved. Taking the second classification as an example, the specific formula is as follows:

$$\mathrm{FL}(u)=\begin{cases}-(1-u)^{\delta}\log(u), & y=1,\\ -u^{\delta}\log(1-u), & \text{otherwise,}\end{cases} \tag{1}$$

where u is the prediction probability of the network to the sample. δ is the added suppression parameter, which is greater than 0 and is used to suppress the weight of easy-to-classify samples. y is the actual category.

For example, when $\delta = 2$, for normal samples, the prediction result of 0.9 is definitely a simple sample. So $(1-0.9)^{\delta}$ will be very small, and then the loss function value will become smaller. However, the sample with a prediction probability of 0.3 has a relatively large loss. The same is true for negative samples. The predicted result of 0.1 should be much smaller than the predicted sample loss of 0.7. Therefore, the training will focus on samples that are difficult to classify.

The SSD algorithm uses the cross entropy loss function as the classification loss function. Hard negative mining is adopted for the sample imbalance problem, ignoring the influence of more negative samples with low loss value on training. The emphasis of network training depends on the loss value gap between them. Compared with the cross entropy loss function, the loss value of difficult samples in Focal Loss is reduced, but the easy-to-classify samples are reduced more. Based on this, the algorithm in this paper integrates the Truncated Gradient idea on the basis of Focal Loss, which makes the training more focused on hard-to-classify samples and makes the training effect better.

3. Animation Characters' Target Detection Algorithm Based on Clustering and Cascaded SSD

3.1. Anime Character Data Set Construction. The training of intelligent recognition model of animation characters requires a large number of animation characters' images and labels of the characters in the images. Table 1 shows some classic animation characters and objects.

Each animation character needs at least 500 pictures of different angles and shapes for the training set. In addition, no less than 30% of the samples in the training set should be prepared as the test set. Pictures of animation characters used as training samples should be jpg, jpeg, bmp, and png. The resolution of the picture is not less than 640 ∗ 480. The training data source flow is shown in Figure 1.

The source data is mainly video and picture data, and the sources are mainly divided into the following categories:

(1) Relevant video and picture data acquired in the content management platform.

(2) Related video and picture data crawled and downloaded from the Internet.

(3) Operators upload relevant video and picture data.

(4) Data enhancement.

3.2. Target Detection Algorithm Based on Clustering and Cascaded SSD. In this paper, the algorithm adopts an improved classification loss function for the bottom detector during training, which makes the training more inclined to samples that are difficult to classify. In this way, a better model for small target position detection can be obtained. The forecast is divided into two stages. In the first stage, the features of each layer of the picture are extracted by the trained model and the targets are detected separately. In this stage, the small target border Pb_s with unknown category and the big target border B_{b} with known specific category can be obtained. In the second stage, the small target enhancement detection module is designed to cascade with the SSD network. In this stage, it is necessary to extract high-level features of Pb_s and then detect them. In this way, a small target detection result B_{s} is obtained. Then non-maximum suppression (NMS) is applied to B_{b} and B_{s}, and the final detection result B_{fr} is obtained. Finally, the k-means clustering algorithm is used to reconstruct the regional candidate box to improve the detection accuracy of the algorithm. The algorithm is mainly divided into four parts, which are small target position detection module, big target detection module, small target enhancement detection module, and k-means clustering algorithm reconstruction region candidate box module. The specific network structure of is shown in Figure 2.

Table 1: List of animation characters and objects.

Content name	Character	Content name	Character	Content name	Character
Prince of Tennis	Ryoma dragon horse	Naruto	Lillock	Inuyasha	Inuyasha
The magic bucket quickly	Criminal kid	Naruto	Yu Zhibo Sasuke	Inuyasha	Platycodon Grandiflorum
Detective Conan	Maori Kogoro.	Dragon ball	Sun WuKong	Inuyasha	Kagome
Detective Conan	Conan Edogawa	Sea king	Robin	Inuyasha	Sesshomaru
Detective Conan	The Maori orchid	Sea king	Nami	Inuyasha	Qibao

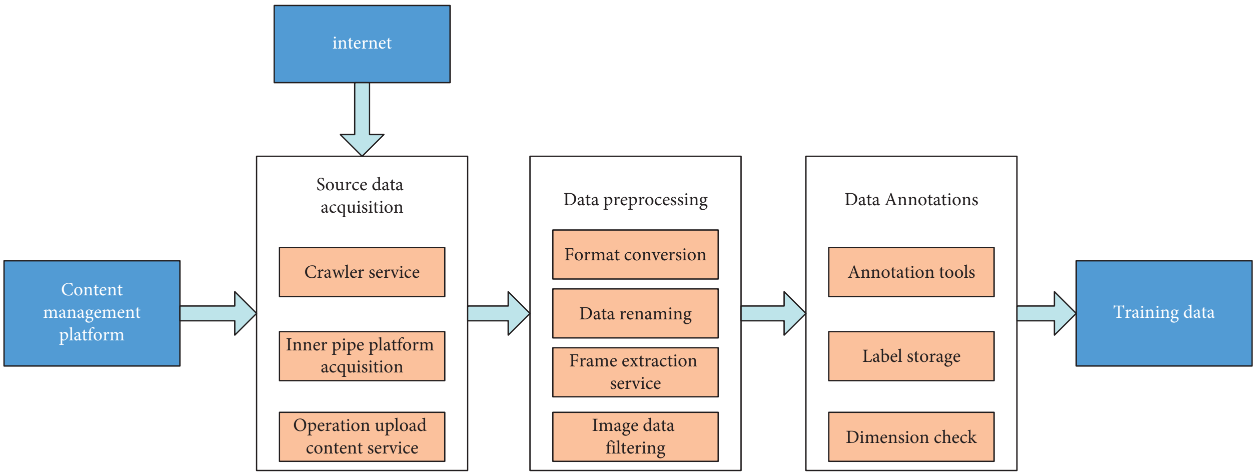

Figure 1: Training data generation flow.

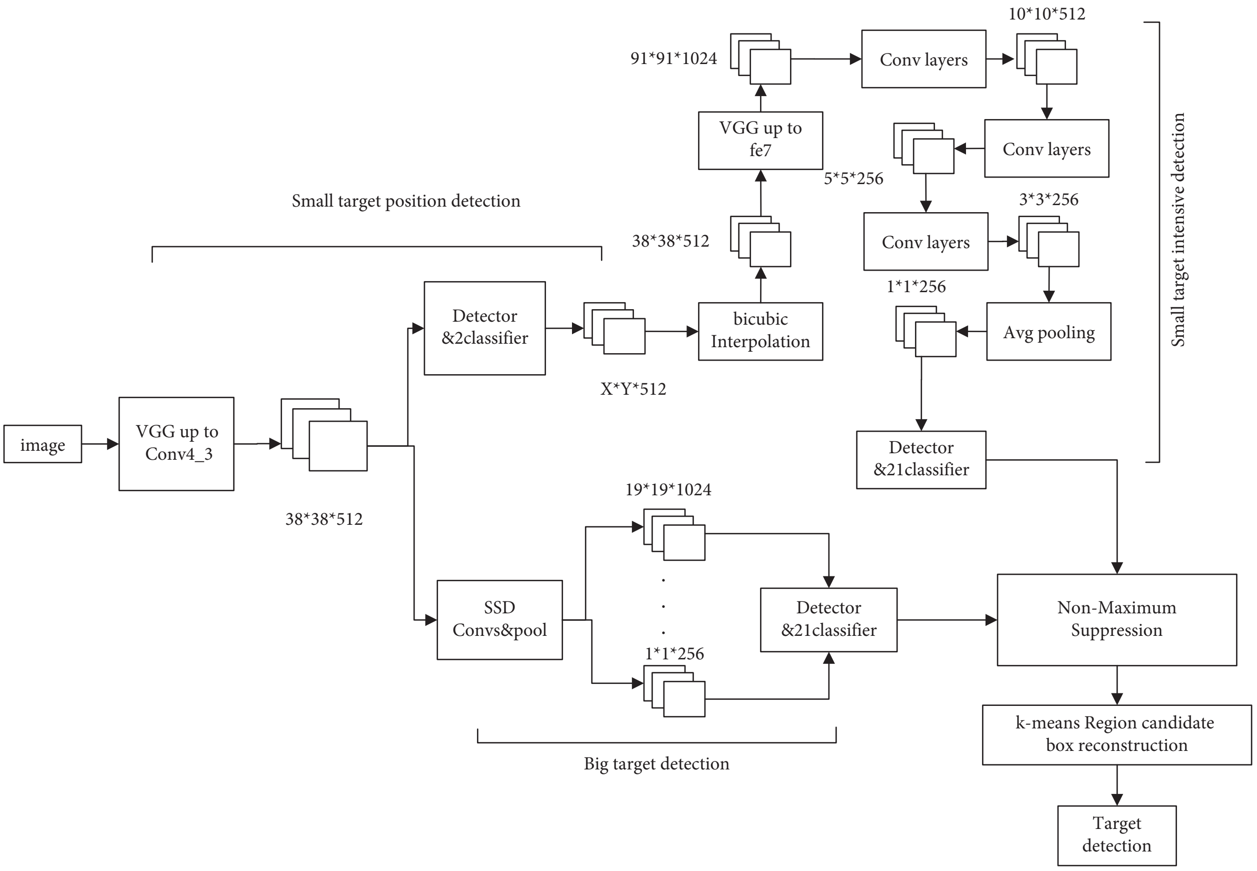

Figure 2: The specific network structure of this algorithm.

3.2.1. Small Target Position Detection Module. At the initial stage of the whole network, a small target position detection module is set up. VGG16 is used as the basic feature extraction network. The image is convolved several times after input to layer Conv4_3. The features of this layer are detected as the bottom feature map of the picture. At this time, the detector used is 2 classification and position regression, which only detects the position of the small target without detecting the specific category of the small target. The border of the small target position is obtained without knowing the specific category. The main tasks of this module are as follows:

(1) Improve the classification loss function Enhanced Focal Loss to replace the original cross entropy loss function and improve the network detection effect.

(2) Design a new small target detector to detect the small target position to improve the network detection effect and reduce the network computation.

In the training stage of the network, the imbalance between easy-to-classify samples and hard-to-classify samples will have a great influence on the training results. Easy-to-classify samples refer to the samples that are easy to be detected by the network. Generally, they refer to the samples with the prediction probability $p > 0.6$. The network prediction probability of the other sample is $p \leq 0.6$, which is difficult to predict accurately. Network learning this kind of fuzzy sample is more difficult, which is called a difficult sample. If samples are not balanced, the difficult samples have little influence on the weights generated in the network learning process. With the network training, difficult samples may be taken as background and then ignored.

The original SSD network adopts the cross entropy loss function as the classification loss function. It uses hard negative mining to select negative samples for the sample imbalance problem. The positive and negative sample ratio is 1 : 3. However, the negative samples are sorted according to the loss value of samples, and the negative samples with larger loss are selected as the negative samples for training. But in this way, negative samples with less loss are ignored. Although the loss of these negative samples is small, the number and the combined loss are large. Based on the ideas of Focal Loss and Truncated Gradient, this paper proposes an improved classification loss function, Enhanced Focal Loss, to replace the original cross entropy loss function. The advantage is that the influence of a large number of negative samples with a small loss value on the training is considered, and the network training is more obviously biased towards samples that are difficult to classify. The function is mainly divided into the following four parts:

(1) Based on the original cross entropy loss function, it describes the distance between the actual output u (the prediction probability of the network to the sample) and the expected output (the actual category of the sample). The smaller the value of cross entropy is, the smaller the loss value is, and the more accurate the network prediction is.

$$\begin{gathered} \mathrm{CE}(u, y) = -\log(u_n), \\ u_n = \begin{cases} u, & \text{if } y = 1, \\ 1 - u, & \text{otherwise}, \end{cases} \end{gathered} \tag{2}$$

where u_n is the overall cross entropy loss function. u is the output probability. y is the actual category.

(2) The equilibrium coefficient β is added to optimize the imbalance of positive and negative samples.

$$\begin{gathered} \mathrm{CE}(u_n) = -\beta_n \log(u_n), \\ \beta_n = \begin{cases} \beta, & \text{if } y = 1, \\ 1 - \beta, & \text{otherwise}, \end{cases} \end{gathered} \tag{3}$$

where β_n is the balance coefficient. β is a balance parameter to be set by yourself, which is used to control the weight of positive and negative samples. y is the actual category of the sample. u_n is the sample output probability.

(3) Focal Loss idea is incorporated and inhibition coefficient $(1 - u_n)^\delta$ is added to make network training slightly biased to hard-to-classify samples.

$$\mathrm{FL}(u_n) = \begin{cases} -\beta(1 - u_n)^\delta \log(u_n), & y = 1, \\ -(1 - \beta)(1 - u_n)^\delta \log(u_n), & \text{otherwise}, \end{cases} \tag{4}$$

where $(1 - u_n)^\delta$ is the suppression coefficient. δ is a suppression parameter to be set by yourself to suppress the weight of easy-to-classify samples. y is the actual category of the sample. u_n is the sample output probability.

(4) Incorporating Truncated Gradient and adding truncation coefficient $\varepsilon(u_n)$. In this way, the influence of a large number of negative samples with a small loss value on the training is considered, which makes the network training greatly biased towards the samples that are difficult to classify.

$$\varepsilon(u_n) = \begin{cases} (1 - u_n)^\delta, & u_n > 0.6, \\ \delta \cdot (1 - u_n), & \text{otherwise}. \end{cases} \tag{5}$$

Formula (5) defines the truncation coefficient as u_n calculating the loss value in sections for the cut-off point. δ is a parameter that needs to be set by yourself to suppress the weight of easy-to-classify samples. u_n is the prediction probability of the network to the sample.

$$\mathrm{EFL}(u_n) = \begin{cases} -\beta \cdot (1 - u_n)^\delta \cdot \log(u_n), & y = 1,\ u_n > 0.6, \\ -\beta \cdot \delta \cdot (1 - u_n) \cdot \log(u_n), & y = 1,\ \leq 0.6, \\ -(1 - \beta) \cdot (1 - u_n)^\delta \cdot \log(u_n), & y \neq 1,\ u_n > 0.6, \\ -(1 - \beta) \cdot \delta \cdot (1 - u_n) \cdot \log(u_n), & y \neq 1,\ u_n \leq 0.6. \end{cases} \tag{6}$$

Formula (6) is a global Enhanced Focal Loss function. β is a balance parameter to be set by yourself, which is used to control the weight of positive and negative samples.

The network can better extract the underlying features of the picture in the prediction stage after using the Enhanced Focal Loss as the loss function. It is input to the detector for detection after extracting the bottom feature map used to detect the small target. The original bottom feature detector is to directly carry out 21 classifications and position regression. Some areas containing targets can be detected in this way in that the location information of the low-level feature map is sufficient. However, due to insufficient semantic information, it is difficult to achieve accurate classification, which will eventually be abandoned. It leads to poor detection effects for small targets. Based on this, this paper designs a small target detector to carry out 2 classifications and position regression. The small target detector only detects the small target area but does not detect the specific category of this small target. Thus, a better detection effect of a small target position is obtained. At the same time, this method changes the classification from 21 to 2, which reduces the network parameters and computation. The specific steps of the detector are as follows:

(1) The original picture x is input into the detection network, and the underlying feature map $\mathrm{FM}_1 = H_4(H_3(\cdots H_1(x))$ for detecting small targets is extracted, where x represents the original drawing. $H_1(x)$ represents the features extracted after the first convolution.

(2) 2-classification detection and position regression are performed on the feature map FM_1, and $pb_s = (\mathrm{Detector}_1(\mathrm{FM}_1))$ is obtained, where FM_1 represents the bottom feature map for detecting small targets, and $\mathrm{Detector}_1$ is a small target detector.

These borders pb_s are subject to nonmaximum suppression, which can suppress some overlapping or incorrect borders. After that, we can get the small target position frame $Pb_s = \mathrm{NMS}(pb_s)$ without knowing the specific category, but with an accurate position, where pb_s represents all small target frames obtained by the small target detector. $\mathrm{NMS}(pb_s)$ means to perform nonmaximum suppression operation on pb_s.

3.2.2. Big Target Detection Module. Based on the idea of the original SSD, big target detection will use multiple detectors to detect on multiple high-level feature maps and get specific types of l big target detection frames. The location information and semantic information of the big target on the high-level feature map are sufficient, and the specific category and location can be directly detected. Therefore, all high-level detectors are 21-classification detection and position regression. The specific detection steps of this module are as follows.

(1) The original image x is input into the detection network, and the bottom feature image FM_1 used to detect small targets is extracted.

(2) The feature map FM_1 is convolved for three times, and the first high-level feature map for detecting big targets is obtained. $\mathrm{FM}_2 = H_7(H_6(H_5(\mathrm{FM}_1)))$, where FM_1 is the bottom feature map. $H_5(\mathrm{FM}_1)$ is the fifth convolution of FM_1.

(3) Similar to the operation in the second step, backward convolution and pooling are continued to obtain the following four feature graphs FM_3, FM_4, FM_5, and FM_6 for detecting big targets.

(4) For these high-level feature maps, their corresponding detectors are used to obtain multiple big target frames $b_b = \mathrm{Detector}_2(\mathrm{FM}_2) + \cdots + \mathrm{Detector}_6(\mathrm{FM}_6)$, where $\mathrm{Detector}_n(\mathrm{FM}_n)$ indicates that the detector corresponding to the feature map is used for detection.

(5) These frames b_b are subjected to nonmaximum suppression processing, and some overlapping or incorrect frames are suppressed so as to obtain a big target frame $B_b = \mathrm{NMS}(b_b)$ with a specific category and accurate position. In which b_b represents a plurality of big target frames obtained by the big target detector. $\mathrm{NMS}(b_b)$ represents nonmaximum suppression operation for b_b.

$$\begin{aligned} b_b &= \mathrm{Detector}_2(\mathrm{FM}_2) + \cdots + \mathrm{Detector}_6(\mathrm{FM}_6) B_b \\ &= \mathrm{NMS}(b_b) B_b = \mathrm{NMS}(b_b). \end{aligned} \tag{7}$$

3.2.3. Small Target Enhancement Detection Module. When small targets are detected on the underlying feature map, the underlying semantic information is not enough to detect specific categories. When detecting on the high-level feature map, the small target with a very small size does not have enough position information among the features with enough high-level semantic information, making it impossible to detect small targets. Based on this, a small target enhancement detection module and SSD cascade are designed to detect the specific category and accurate location of the small target frame B_s.

The module is divided into two parts. The first part is to transform small target detection into big target detection. The bottom feature of the small target location is sampled to the size of FM_1 of the bottom feature of the original image, so the small target detection problem is transformed into a big target detection problem. The second part is to extract high-level features and detect them and learn the advantages of the original SSD network in detecting big targets. The subsequent stage of this module is to extract high-level features from the upsampled feature map and then detect it. At this time, the network layer similar to the original SSD network is used to extract high-level features, and the last high-level features are used for target detection. The main steps of this module are as follows:

(1) The small target position frame Pb_s obtained from the small target position detection module is input. Based on the principle of translation invariance of CNN (Convolutional Neural Network), the bottom

feature FM_s of the position corresponding to pbs is obtained from the bottom feature map FM_1.

(2) These small target bottom features FM_s are upsampled by a bicubic interpolation algorithm, and the small target bottom feature map BFM_s with the same size as the original bottom feature map FM_1 is obtained.

$$\mathrm{BFM}_s = \mathrm{FM}_s \uparrow s, \tag{8}$$

where $\uparrow s$ represents the upsampling operator.

(3) Then, the low-level feature map BFM_s of small targets is convolved and pooled for many times, its high-level feature UFM_s is extracted. The high-level feature UFM_s is classified by 21 and its position is detected, so as to obtain a plurality of small target frames b_s.

$$\begin{aligned} \mathrm{UFM}_s &= P\left(H_{10}\left(H_9\left(\cdots H_5\left(\mathrm{UFM}_s\right)\right)\right)\right), \\ b_s &= \mathrm{Detector}_6\left(\mathrm{UFM}_s\right), \end{aligned} \tag{9}$$

where $P(\cdot)$ represents the average pooling operation. $H(\cdot)$ represents the convolution operation. $\mathrm{Detector}_6$ is the detector.

(4) Finally, these frames b_s are suppressed by non-maxima. Some overlapping or incorrect frames are suppressed, so as to obtain the big target frames B_s with specific categories and accurate positions.

$$B_s = NMS\left(b_s\right), \tag{10}$$

where b_s represents a plurality of small target frames detected from the high-level features of small targets. $\mathrm{NMS}(b_s)$ represents nonmaximum suppression operation for b_s.

3.2.4. Reconstruction of Regional Candidate Box Based on k-Means Clustering. The performance of a deep learning target detection algorithm largely depends on the quality of feature learning. Feature learning is driven by training data, which in the SSD target detection task is the area candidate box. When the intersection union ratio (IOU) of the region candidate frame and the real frame is greater than the threshold, it is marked as a positive sample. On the contrary, it is marked as a negative sample. The formula of candidate box parameters is as follows:

$$F_z = F_{\min} + \frac{F_{\max} - F_{\min}}{w-1}(z-1), \tag{11}$$

where the value range of z is $[1, w]$. w is the characteristic layer number. $F_{\min}$ and $F_{\max}$ represents the minimum and maximum characteristic layer scale, respectively, and the corresponding default values are generally 0.2 and 0.9. The middle characteristic layer is evenly distributed in scale. As shown in Figure 3, the region candidate algorithm based on clustering can generate four candidate boxes, including two squares (red dotted lines) and two rectangles (black dotted lines).

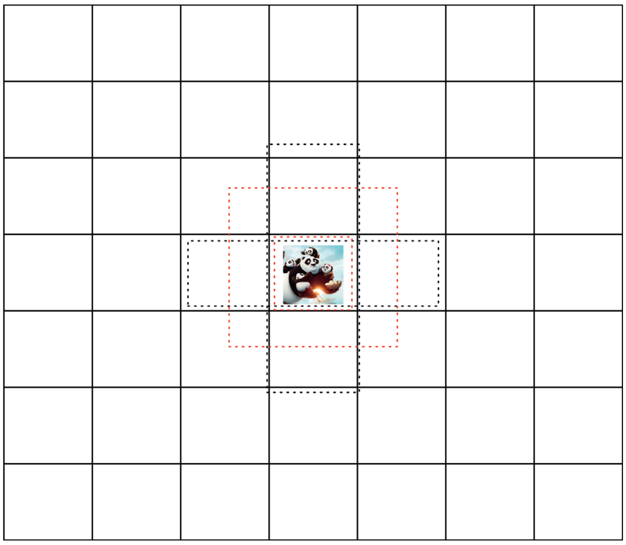

Figure 3: Schematic diagram of candidate box.

In the past tests, most of them set the size and proportion of candidate boxes according to experience. During the training process, the network will adjust the candidate box. If the appropriate size and proportion of candidate boxes can be found in advance before training, better prediction results will be obtained. In this paper, the k-means clustering algorithm is used to predict the size and proportion of candidate frames. The distance measurement formula used in this paper is as follows:

$$\begin{aligned} &d(\text{box}, \text{center}) = 1 - IOU(\text{box}, \text{center}), \\ &d(\text{box}, \text{center}) = 1 - IOU\left[\left(x_q, y_q, w_q, h_q\right), \left(x_p, y_p, W_p, H_p\right)\right], \end{aligned} \tag{12}$$

where center is the clustering center. *box* is the callout box. When calculating, the value of IOU can only be calculated if the center point of each annotation box coincides with the center point of the cluster center. (x_q, y_q) is the center point of the callout box. (w_q, h_q) is the width and height of the callout box. (W_p, H_p) is the width and height of the cluster center. $q \in \{1, 2, \ldots\ldots N\}$, $p \in \{1, 2, \ldots\ldots K\}$. N is the number of annotation boxes. K is the number of clusters. According to the above formula, the annotation box is assigned to the nearest cluster center. After all annotation boxes are allocated, the cluster center points are recalculated for each set.

$$\begin{aligned} W_i' &= \frac{1}{N_i}\sum_{w_i}, \\ H_i' &= \frac{1}{N_i}\sum h_i. \end{aligned} \tag{13}$$

Find the average width and height of all annotation boxes in the set, and then repeat the above steps until the cluster center changes little. In the experiment, $K = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ is set for the experiment, and the marked boxes were clustered separately. The experimental results show that when $K < 6$, the average intersection ratio increases greatly. When $K > 6$, it tends to be flat basically. Combined with the algorithm, $K = 6$ is selected. When $k = 6$, the aspect ratio (AR) of candidate frames is [0.59, 0.89, 1.18, 1.84, 1.9, 2.84]. Reconstruct the candidate box with this AR.

Cascade SSD algorithm uses six feature maps to generate candidate boxes with different sizes. If the size of the feature graph is $M * N$, the corresponding feature graph is divided into $M * N$ grids. Then H candidate boxes are generated with the center point of each grid as the center. The number of candidate boxes generated by each feature graph is $M * N * H$. In this algorithm, the AR of six feature maps is set to {1, 2, 3}. The number of H is {6, 6, 6, 6, 6, 6}. Table 2 shows the setting of the candidate box of the algorithm area.

When the algorithm is used for target detection in this paper, the final detection result is related to the prediction of the regional candidate box. However, the regression of the prediction frame of the regional candidate has a great relationship with the samples in a certain range around it. Therefore, an appropriate threshold value should be selected when reconstructing the region candidate box. Experiments were carried out under different thresholds to select a suitable threshold. The experimental results show that the optimal threshold is 0.45.

4. Experiment and Analysis

4.1. Experimental Environment and Data Set. The operating system of the experimental software environment is Ubuntu14.04, using the TensorFlow deep learning framework. The training environment settings are shown in Table 3.

The animation character material pictures used for detection can be obtained from the screenshots in the video library, with abundant sources and a large amount of data. 80% of the training set and 20% of the test set are used to divide the training set and testing set. The animation character material pictures are obtained by searching the cosplay pictures of specific characters from the Internet. The amount of data is meager. Therefore, training and test sets are divided into 85% training set and 15% test set during training.

4.2. Evaluation Criteria. In this paper, mAP (mean average precision) is adopted to evaluate the detection performance of the algorithm. The detection speed is used to evaluate the detection speed of the algorithm, and its unit is fps. Detection speed refers to the number of pictures that can be processed per second. Detection speed comparison needs to be done on the same hardware. mAP refers to the accuracy of all kinds of objects on all graphs, and its calculation formula is as follows:

$$\mathrm{mAP} = \frac{\sum \mathrm{Precision}_{\mathrm{Aver}}}{N}, \tag{14}$$

where $\mathrm{Precision}_{\mathrm{Aver}} = \sum \mathrm{Precision}_C / N_{\mathrm{Images}}$ represents the average precision sum of all categories. N means all categories.

4.3. Experimental Analysis. In order to verify the detection effect of this algorithm, experiments will be carried out from the following aspects. To verify that the loss function Enhanced Focal Loss designed in this paper is more effective than the cross entropy loss function and Focal Loss in this algorithm, these three loss functions are applied to the training of this algorithm, respectively. The selection of training samples, function composition, and comparison of detection accuracy of the three functions are shown in Table 4.

As shown in Table 4, the original cross entropy loss function selects positive samples and negative samples with large loss values as training samples. In this way, a large number of negative samples with a small loss value are lost, which is obviously unfavorable to network training. Finally, the detection accuracy of this algorithm is 78.73%. Focal Loss selects all the samples as training samples, including a large number of negative samples with small loss values, and added suppression coefficient to control the loss of hard-to-classify samples and easy-to-classify samples. This makes the loss of easy-to-classify samples greatly reduced, and the loss of difficult-to-classify samples slightly reduced. On the whole, the weight of hard-to-classify samples increased slightly, which made the network training biased towards hard-to-classify samples. Finally, the detection accuracy of Focal Loss in this algorithm is 78.8%. The Enhanced Focal Loss used in this paper is integrated with Truncated Gradient thought, which makes the loss of easily classified samples decrease. The loss of hard-to-classify samples increases and the overall weight of hard-to-classify samples increases significantly, making network training more obviously biased towards hard-to-classify samples. Finally, the detection accuracy of this algorithm is 78.94%.

Some existing SSD improved algorithms are trained and predicted on relevant data sets, and the training and prediction results are compared with the target detection algorithm in this paper. The comparison results are shown in Table 5.

Analysis of Table 5 shows that compared with the improved SSD algorithm, the detection accuracy of the improved algorithm proposed in this paper is obviously improved. The real-time detection performance of the algorithm is well maintained. Especially, the detection speed of this algorithm is nearly three times faster than DSSD.

Some existing target detection algorithms are trained and predicted on the experimental design data set. The training and prediction results are compared with the target detection algorithm in this paper. The comparison results are shown in Figure 4.

As can be seen from Figure 4, the accuracy of this paper in animation character detection is larger than other algorithms. The predicted mAP of the proposed algorithm is 79.7%, which is 15.7% higher than that of YOLO. That's a 2.1% increase over DSSD. It can be verified that in the object detection of cartoon characters, the clustering and cascaded SSD algorithm proposed in this paper not only has a faster detection speed but also significantly improves the detection quality.

Table 2: Setting of candidate box of algorithm area in this paper.

Characteristic diagram	Conv4_3	Fc7	Conv6_2	Conv7_2	Conv8_2	Conv9_2
M∗N	38 ∗ 38	19 ∗ 19	10 ∗ 10	5 ∗ 5	3 ∗ 3	1 ∗ 1
Quantity	8664	2166	600	150	54	6
Total	8664 + 2166 + 600 + 150 + 54 + 6 = 11640					

Table 3: Configuration of training environment.

Serial number	Package	Model
1	CPU	Intel(R) Xeon(R) CPU E5-2620 v3 2.4 GHz 12 core
2	Internal storage	128 GB
3	Hard disc	2 ∗ 200 GB SSD + 2 ∗ 1.2 TB SAS
4	Display card	NVIDIA (R)GTX (R)1080TI
5	System kernel	Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0–135-generic x86_64)

Table 4: Comparison of loss functions.

Loss function	Positive samples	Negative samples (big loss)	Negative samples (small loss)	Inhibition coefficient	Truncated gradient	mAP (%)
Cross entropy	✓	✓	✗	✗	✗	78.73
Focal loss	✓	✓	✓	✓	✗	78.8
Enhanced focal loss	✓	✓	✓	✓	✓	78.94

Table 5: Comparison between MPa and detection speed of SSD-related improved algorithms.

Target detection algorithm	Training MAP (%)	Predication MAP (%)	Detection speed (fps)
Method in [15]	76.1	74.8	54
Method in [16, 17]	74.7	73.9	58
DSSD	74.1	72.6	19
RSSD	76.3	75.1	58
FSSD	74.3	73.7	56
Our method	78.4	76.5	59

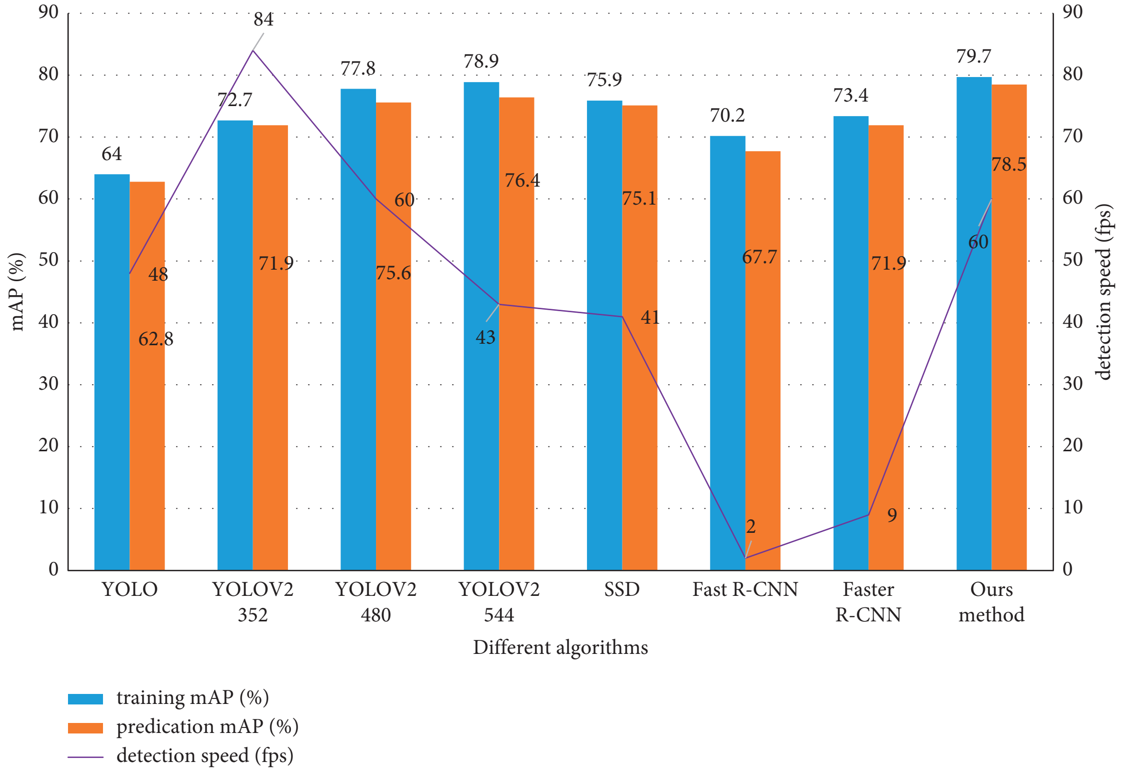

Figure 4: Comparison of target detection algorithms MPa and detection speed.

5. Conclusion

Aiming at the problem of animation character detection, this paper proposes a novel method based on clustering and cascaded SSD. A new loss function is designed to solve the problem of unbalanced training samples for small targets. The algorithm extracts the features of each layer of the image through the trained model and detects the target, respectively. In this way, the small target border of an unknown specific category and the big target border of the known specific category can be obtained. Then, a small target enhancement detection module, which is cascaded with an SSD network, is designed by the algorithm. Additionally, the high-level features corresponding to the small target area can be extracted separately to detect small targets. It can be seen that this algorithm is superior to other existing algorithms via the final experimental comparison. The future work is to improve the upsampling method and enhance the positioning ability of small targets.

Abbreviations

abrIP:	Intellectual property
SSD:	Single shot multibox detector.

Data Availability

The labeled dataset used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The author declares no competing interest.

Acknowledgments

This paper was supported byXijing University Research Fund “one belt one road directive” under the background, Innovative design of cartoon image of door God in Shanxi Province of Fengxiang Province” (No. XJ160207).

References

[1] M. Pellitteri, “The European experience with Japanese animation, and what it can reveal about the transnational appeal of anime,” *Asian Journal of Communication*, vol. 31, no. 1, pp. 21–42, 2021.

[2] K. Liu and X. Q. Sun, “Research on the development and innovation of animation industry in jilin Province in the Internet big data era,” in *Proceedings of the IOP Conference Series: Earth and Environmental Science*, vol. 619, no. 1, Article ID 012073, Changchun, China, August 2020.

[3] J. H. Yan, B. C. Lee, and T. Yun, “A study on the elements of Chinese animation IP (intellectual property) development based on the pan-entertainment industry,” *International Journal of Internet, Broadcasting and Communication*, vol. 13, no. 1, pp. 168–179, 2021.

[4] J. Zhang, X. Hu, Z. Ning et al., “Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2633–2645, 2018.

[5] P. Sun, R. Zhang, Y. Jiang et al., “Sparse r-cnn: end-to-end object detection with learnable proposals,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2021, Article ID 14454.

[6] Z. Y. Chen and I. Y. Liao, “Improved fast r-cnn with fusion of optical and 3d data for robust palm tree detection in high resolution uav images,” *International Journal of Machine Learning and Computing*, vol. 10, no. 1, pp. 122–127, 2020.

[7] Z. Ning, X. Hu, Z. Chen et al., “Obaidat, A cooperative quality-aware service access system for social Internet of vehicles,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2506–2517, 2017.

[8] Z. Shao, W. Zhou, X. Deng, M. Zhang, and Q. Cheng, “Multilabel remote sensing image retrieval based on fully convolutional network,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 318–328, 2020.

[9] L. Zhao and S. Li, “Object detection algorithm based on improved YOLOv3,” vol. 9, no. 3, pp. 537–548, 2020.

[10] L. Fu, Y. Feng, J. Wu et al., “Fast and accurate detection of kiwifruit in orchard using improved YOLOv3-tiny model,” *Precision Agriculture*, vol. 22, no. 3, pp. 754–776, 2021.

[11] C. Chakraborttii and H. Litz, “Improving the accuracy, adaptability, and interpretability of SSD failure prediction models,” in *Proceedings of the 11th ACM Symposium on Cloud Computing*, pp. 120–133, ACM, Times Square, NY, USA, October 2020.

[12] B. Hu, G.-P. Gao, LeLe He, X.-D. Cong, and J.-N. Zhao, “Bending and on-arm effects on a wearable antenna for 2.45 GHz body area network,” *IEEE Antennas and Wireless Propagation Letters*, vol. 15, pp. 378–381, 2016.

[13] Y. Yang, J. Yu, and T. Kurihara, “Immature Yuzu Citrus Detection Based on DSSD Network with Image Tiling approach,” in *Proceedings of the 2021 60th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 1128–1133, IEEE, Tokyo, Japan, September 2021.

[14] S. Wang, R. Inkol, and B. R. Jackson, “Relationship between the maximum likelihood emitter location estimators based on received signal strength (RSS) and received signal strength difference (RSSD),” in *Proceedings of the 2012 26th Biennial Symposium on Communications (QBSC)*, pp. 64–69, IEEE, Kingston, ON, Canada, May 2012.

[15] J. W. Wen, Y. W. Zhan, C. H. Li, and J. Lu, “Design of atrous filter to strengthen small object detection capability of SSD,” *Application Research of Computers*, vol. 36, no. 3, pp. 861–865, 2019.

[16] X. Hu, J. Cheng, M. Zhou et al., “Emotion-Aware cognitive system in multi-channel cognitive radio ad hoc networks,” *IEEE Communications Magazine*, vol. 56, no. 4, pp. 180–187, 2018.

[17] W. Wei, B. Zhou, D. Polap, and M. Wozniak, “A regional adaptive variational PDE model for computed tomography image reconstruction,” *Pattern Recognition*, vol. 92, pp. 64–81, 2019.