

Multi target pigs tracking loss correction algorithm based on Faster R-CNN

Longqing Sun^{1*}, Yuanbing Zou¹, Yan Li², Zhengda Cai³, Yue Li¹, Bing Luo¹, Yan Liu¹, Yiyang Li¹

(1. College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China; 2. Kunming Dacun Sci-Tech Co., Ltd, Kunming 650000, Yunnan; 3. Yunnan Provincial Academy of Science and Technology, Kunming 650000, Yunnan)

Abstract: In order to solve the problem that target tracking frames are lost during the visual tracking of pigs, this research proposed an algorithm for multi target pigs tracking loss correction based on Faster R-CNN. The video of live pigs was processed by Faster R-CNN to get the object bounding box. Then, the SURF and background difference method were combined to predict whether the target pig will be occluded in the next frame. According to the occlusion condition, the maximum value of the horizontal and vertical coordinate offset of the bounding box in the adjacent two frames of the frame image in continuous N (N is the value of the video frame rate) were calculated. When bounding boxes in a video frame are merged into one bounding box, this maximum value was used to correct the current tracking frame offset in order to achieve the purpose of solving the tracking target loss problem. The experiment results showed that the success rate range of RP Faster-RCNN in the data set was 80%-97% while in term of Faster-RCNN was 40%-85%. And the average center point error of RP Faster-RCNN was 1.46 lower than Faster-RCNN which was about 2.60. The new algorithm was characterized by good robustness and adaptability, which could solve the problem of missing tracking target and accurately track multiple targets when the targets occlude each other.

Keywords: object tracking, Faster-RCNN, individual pig, target occlusion

DOI: 10.25165/ijabe.20181105.4232

Citation: Sun L Q, Zou Y B, Li Y, Cai Z D, Li Y, Luo B, et al. Multi target pigs tracking loss correction algorithm based on Faster R-CNN. Int J Agric & Biol Eng, 2018; 11(5): 192–197.

1 Introduction

China is a country with the largest number of pigs breeding and the most pork consumption. With the continuous expansion of the large scale and intensification, the breeding risk also gradually increased. Tracking and recording individual behavior of pigs contributes to detect the abnormal behavior of pigs in advance and reduces the incidence of pig diseases^[1-3]. The traditional method of manually recording the target behavior of pigs causes the problem of recording data with large amount of errors and consuming a lot of manpower. It improves the accuracy and real-time of data collection to Adopt machine vision target tracking technology to replace manual collection and analysis of pig behavior information. It helps for discovering abnormal behavior of pigs in advance, and taking appropriate measures in time to reduce the occurrence of pig diseases^[4,5].

Target tracking is one of the hot issues in the field of machine vision target tracking^[6,7]. Through analyzing the characteristics of

a particular target, the target position and the change of the region in the video sequence are obtained, which facilitates the subsequent data analysis, pattern recognition and other applications. The Convolution Neural Network target tracking method based on strong learning ability, the advanced ability of semantic information acquisition and efficient feature expression, are helpful for automatically extracting the target features and obtaining more effective representation of the image features^[8-10]. Zhou et al.^[11] proposed a method of identifying the main organs of tomato based on deep Convolutional Neural Network, and it realized rapid and accurate detection of different organs of tomato. Ma et al.^[12] proposed a target tracking algorithm based on the hierarchical convolution feature. The algorithm used the convolution feature and the hierarchical recursion idea to determine the target position, which significantly improved the performance of the target tracking algorithm. Many researchers have done a lot of research on the challenge of target recognition rate and accuracy reduction due to the change of the target features caused by the target occlusion and ambient light changes^[13,14]. Hua et al.^[15] proposed a method of combining occlusion and motion reasoning with one by one detection method, which could better handle the occlusion problem. Yang et al.^[16] proposed a method through the geometric transformation of the object. It established a rich bounding box set to predict the position of the object and improved the accuracy of the target tracking. Although the above methods have a very good tracking effect in complex environments, it still cannot meet the requirements of high detection accuracy. Girshick et al.^[17] proposed R-CNN. R-CNN extracted about 2000 bounding boxes from the original image through the Selective Search algorithm, scales the bounding boxes to a fixed size, classified the bounding boxes through the CNN network, and it can accurately determine whether the target was to be identified. In response to the slow

Received data: 2018-03-07 **Accepted date:** 2018-07-14

Biographies: **Yuanbing Zou**, Master candidate, research interest: video image processing technology, Email: zourose123@126.com; **Yan Li**, Bachelor, research interest: IoT application, Email: 13888892809@163.com; **Zhengda Cai**, Senior Engineer, research interests: energy saving technology and IoT application, Email: 1294086726@qq.com; **Yue Li**, Master candidate, research interests: video image processing technology, Email: liyuelucky1@126.com; **Bing Luo**, Master candidate, research interests: video image processing technology, Email: 330257373@qq.com; **Yan Liu**, Master candidate, research interests: video image processing technology, Email: 912034741@qq.com; **Yiyang Li**, Master candidate, research interests: video image processing technology, Email: adelally@sina.com.

***Corresponding author:** **Longqing Sun**, Professor, research interest: computer technology. China Agricultural University, Beijing 100083, China. Email: sunlq@cau.edu.cn.

detection of R-CNN, Ren et al.^[18] proposed Faster R-CNN. The method generated region proposals by constructing a region proposal network (RPN), and the appearance of RPN replaced the Selective Search^[19] method in RCNN to improve the proposed region detection efficiency. In order to predict the target position accurately and efficiently, He et al.^[20] proposed a new SURF-based method to describe the relationship between the local feature motion and the object global motion based on SURF's rapid search of image feature points and feature point matching. This method can effectively learn the parameters through the online EM algorithm and adopt the updating mechanism to adapt the object representation. Background difference method is relatively simple and tracking real-time. Takita et al.^[21] used background difference method to achieve real-time and efficient target tracking.

In order to solve the problem that the target tracking frame is lost and the detection efficiency of the tracking algorithm is low due to the mutual occlusion of the targets in the pig visual tracking process, this paper proposes a multi target pigs tracking loss correction algorithm based on Faster R-CNN which uses deep compressing^[22] to accelerate operation. Firstly, the video of live pigs is processed by Faster R-CNN to get the bounding box. Secondly, predict whether the target pig will be occluded in the next frame. Finally, the algorithm proposed in this paper is used to process the position offset of the bounding box in the tracking image according to the occlusion situation, so as to solve the problem of the loss of the tracking box.

2 Materials and methods

2.1 Sample acquisition and marking

Experimental data were collected from 10 pens and 2 videos were collected for each pens, for a total of 20 videos. Each video duration is about 30 min, and the video frame rate is 25 fps, and the image resolution is 600×400 pixels. For each pen house, randomly select 1 video as the test sample and the remaining video as the training sample. There are 10 test videos and 10 training videos in total. In order to ensure good generalization of the training results, the training set of this paper is derived from two parts: 10 000 pictures in 10 training videos and 2000 pictures of pigs from ImageNet data set, totaling 12 000 pictures.

In order to reduce the amount of computation, this paper uses the labelImg, which is an open source image labeling tool to mark the training samples. Pig sample tag image shown in Figure 1.

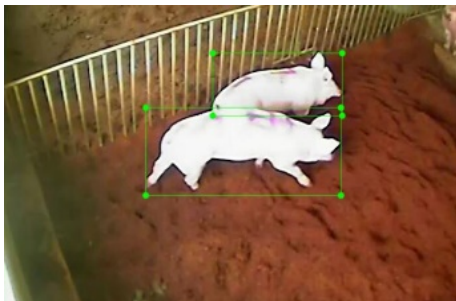


Figure 1 Pig sample tag image

Then do pre-processing, the labeled image is cropped automatically to get pictures. Since this article uses the ZF network structure, these images are sized to 224×224.

2.2 Improved algorithm

2.2.1 Occlusion prediction

(1) Calculate the tracking target's overall movement distance

Because SURF algorithm has very good robustness and high computational speed, this research uses this algorithm to extract the

target characteristic points^[23]. After extracting the feature points and the feature description, it is necessary to match the feature points between the images. In this paper, we use the nearest neighbor nearest neighbor ratio method proposed by Lowe^[24] to match the feature points. In order to prevent too many false matches, this article chooses a ratio of 0.6. The matching effect is shown in Figure 2.

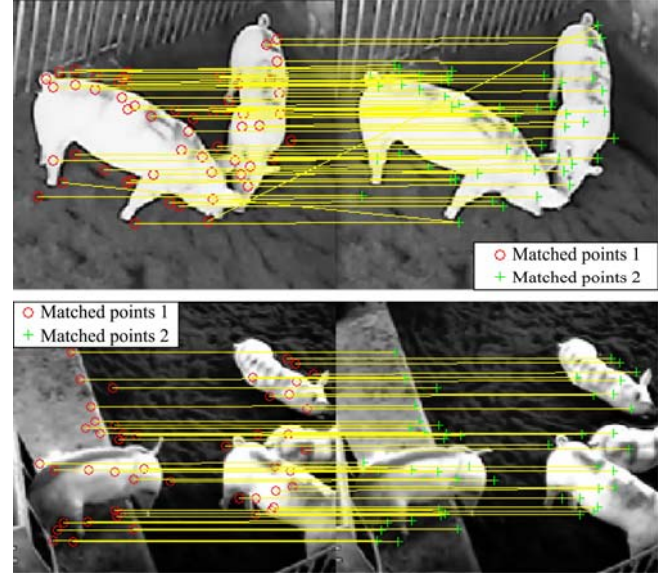


Figure 2 Pig matching map

In this paper, the coordinate difference between the x and y directions of the feature points matching the images are used as the moving distance of the target in the continuous frame. The expression is as follows:

$$(\Delta x_{r,p,j}, \Delta y_{i,k,j}) = T_{r,p,j}(x, y) - T_{r,p,j-1}(x, y) \quad (1)$$

$$(r = 1, \dots, n; p = 1, \dots, m(r))$$

$$\overline{\Delta x_{r,j}} = \frac{1}{m(r)} \sum_{j=1}^{m(r)} \Delta x_{r,p,j} \quad (2)$$

$$\overline{\Delta y_{r,j}} = \frac{1}{m(r)} \sum_{j=1}^{m(r)} \Delta y_{r,p,j}$$

where, $T_{r,p,j}(x, y)$ is the position coordinate of the p^{th} feature point of the r^{th} participating pig target in the j^{th} frame of image; $m(r)$ is the number of matching successful pig target feature points for the r^{th} match; $\Delta x_{r,p,j}$ and $\Delta y_{r,p,j}$ are respectively the x -direction and y -direction moving distance of the p^{th} feature point of the r^{th} participating pig target in the j^{th} frame image; $\overline{\Delta x_{r,j}}$ and $\overline{\Delta y_{r,j}}$ are respectively the average movement distance of the r^{th} target piggyback point in the j^{th} frame image. The position coordinate difference between the position coordinate of the p^{th} feature point in the r^{th} participation matching pig target of the j^{th} frame image and the feature point corresponding to the target image in the $(j-1)^{\text{th}}$ frame may be expressed as that the pig target moves from the current frame to the next frame of moving distance.

The SURF algorithm is used to get some feature points of the tracking target, and then the feature points are matched by the nearest neighbor ratio method. The result calculated by Equation (2) is taken as the moving distance of the whole tracking target.

(2) Determine if the target has an occlusion

In order to predict the target's movement trend and judge whether the target will block each other. This paper proposes to use the combination of SURF feature point detection and background subtraction to predict whether the areas of any two

targets coincide with each other. And if there is a coincidence, it can be considered as a mutual occlusion to the tracking target. It is assumed that the r^{th} target coordinate in the j^{th} binary image is $BL(x_{r,j}, y_{r,j})$ and the area is $BLS_{r,j}$. The area can be calculated from the number of pixels in the area. By Equation (2), it can be predicted that the r^{th} target coordinate in the $(j+1)^{\text{th}}$ frame binary image is $BL(x_{r,j+1}, y_{r,j+1})$.

$$BL(x_{r,j+1}, y_{r,j+1}) = BL(x_{r,j} + \overline{\Delta x_{r,j}}, y_{r,j} + \overline{\Delta y_{r,j}}) \quad (3)$$

$$BLS_{r,q,j} = BLS_{r,j} \cap BLS_{q,j} \quad (r \neq q) \quad (4)$$

where, $BLS_{r,j}$ and $BLS_{q,j}$ are respectively the area of the r^{th} and q^{th} tracking targets in the j^{th} frame of binary image; $BLS_{r,q,j}$ is the area of the overlapping part of the r^{th} and q^{th} tracking target in the j^{th} frame of binary image. If $BLS_{r,q,j} > 0$, that means the r^{th} and q^{th} tracking target in the j^{th} frame appear occluded.

The algorithm steps are divided into the prediction stage and the correcting tracking stage. The position of the prediction stage in the overall framework of the algorithm, as shown in Figure 3.

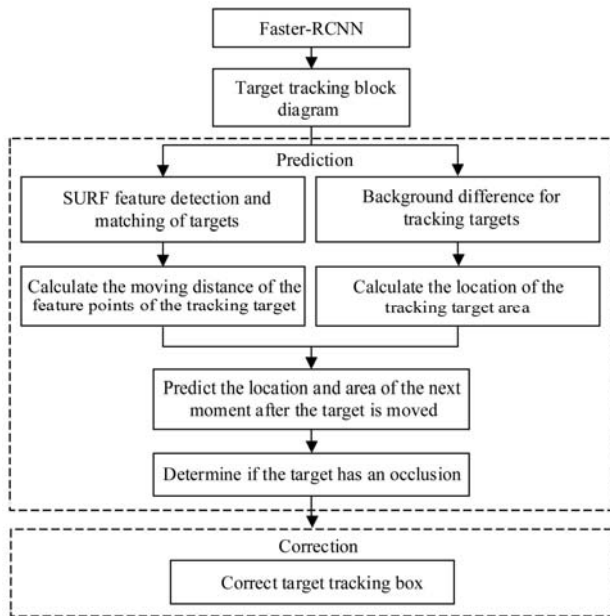


Figure 3 Framework diagram of algorithm

2.2.2 Corrected the target tracking box

In the actual production environment, the characteristics of the pigs are similarity, such as the color of pigs. If the moving objects such as the color of the target features are more similar, a number of targets close to each other are more prone to chaos, loss of tracking and other phenomena. It is easy to make mistakes in the recording of activities such as feeding, drinking and other activities. To solve this problem, we propose a tracking target correction algorithm Revise Position Faster R-CNN (RP Faster R-CNN) based on Faster R-CNN.

With Faster R-CNN training, the tracking target circumscribed rectangles offset after bounding-box regression is taken as input. Assuming that the total number of tracking targets is I , each tracking target circumscribed rectangle is determined by the coordinates of two vertices of the diagonal. According to Equation (4) to judge the target position in the target rectangle.

Supposing that $\{(xc_{1,i,j}, yc_{1,i,j}), (xc_{2,i,j}, yc_{2,i,j})\}$ is the set of two vertices of the diagonal of the i^{th} tracking frame in the j^{th} frame of image. $xc_{1,i,j}$ and $yc_{1,i,j}$ are the coordinate offsets in the x -direction and the y -direction of one vertex respectively. According to analysis, the relationship between the previous frames that are merged by two tracking frames can be divided into two kinds,

respectively, Figure 4a and Figure 4b.

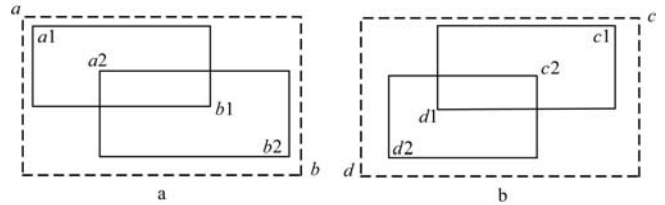


Figure 4 Bounding-box fusion diagram

The solid two-line box in Figure 4a shows two check boxes before merging, and the dashed box shows the merged effect box. Figure 4b is similar to Figure 4a. In Figure 4a, a and b denote the offsets $\{(xc_{1,i,j}, yc_{1,i,j}), (xc_{2,i,j}, yc_{2,i,j})\}$ of the two vertex coordinates of the diagonal of the i^{th} tracking frame in the j^{th} frame of the current frame respectively; $a1$ and $b1$ denote the offsets $\{(x_{1,i,j-1}, y_{1,i,j-1}), (x_{2,i,j-1}, y_{2,i,j-1})\}$ of the two vertex coordinates of the diagonal of the i^{th} tracking frame in the $(j-1)^{\text{th}}$ frame of the current frame respectively; $a2$ and $b2$ denote the offsets $\{(x_{1,k,j-1}, y_{1,k,j-1}), (x_{2,k,j-1}, y_{2,k,j-1})\}$ of the two vertex coordinates of the diagonal of the k^{th} tracking frame in the $(j-1)^{\text{th}}$ frame of the current frame respectively; In Figure 4b, c and d denote the offsets $\{(xc_{2,i,j}, yc_{1,i,j}), (xc_{1,i,j}, yc_{2,i,j})\}$ of the two vertex coordinates of the diagonal of the i^{th} tracking frame in the j^{th} frame of the current frame respectively; $c1$ and $d1$ denote the offsets $\{(x_{2,i,j-1}, y_{1,i,j-1}), (x_{1,i,j-1}, y_{2,i,j-1})\}$ of the two vertex coordinates of the diagonal of the i^{th} tracking frame in the $(j-1)^{\text{th}}$ frame of the current frame respectively; $c2$ and $d2$ denote the offsets $\{(x_{2,k,j-1}, y_{1,k,j-1}), (x_{1,k,j-1}, y_{2,k,j-1})\}$ of the two vertex coordinates of the diagonal of the k^{th} tracking frame in the $(j-1)^{\text{th}}$ frame of the current frame respectively.

In order to judge whether there is a fusion of tracking frames, this paper uses the Manhattan distance^[25] to calculate the distance between two vertices. Assuming that the Manhattan distance from point $x0$ to point $x1$ is denoted as $d(x0, x1)$, the specific judgment method is as follows.

1) According to Equation (4), If $BLS_{r,q,j} > 0$ and $x_{1,i,j-1} \geq x_{1,k,j-1}$, where $r=1, q=k$, the type of fusion of the two tracking frames belongs to Figure 4a. If the Manhattan distance $d(a, a1)$ from point a to point $a1$ is smaller than the Manhattan distance $d(a, a2)$ from point a to point $a2$, and the Manhattan distance $d(b, b1)$ from point b to point $b1$ is greater than the Manhattan distance $d(b, b2)$ from point b to point $b2$, that indicates that the two tracking frames have been merged, otherwise the two tracking frames are not merged. The expressions of $d(a, a1)$, $d(a, a2)$, $d(b, b1)$ and $d(b, b2)$ are as follows.

$$\begin{aligned} d(a, a1) &= |xc_{1,i,j} - x_{1,i,j-1}| + |yc_{1,i,j} - y_{1,i,j-1}| \\ d(a, a2) &= |xc_{1,i,j} - x_{1,k,j-1}| + |yc_{1,i,j} - y_{1,k,j-1}| \\ d(b, b1) &= |xc_{2,i,j} - x_{2,i,j-1}| + |yc_{2,i,j} - y_{2,i,j-1}| \\ d(b, b2) &= |xc_{2,i,j} - x_{2,k,j-1}| + |yc_{2,i,j} - y_{2,k,j-1}| \end{aligned} \quad (5)$$

2) According to Equation (4), If $BLS_{r,q,j} > 0$ and $x_{1,i,j-1} < x_{1,k,j-1}$, where $r=1, q=k$, the type of fusion of the two tracking frames belongs to Figure 4B. Analytical methods are the same as above. The expressions of $d(c, c1)$, $d(c, c2)$, $d(d, d1)$ and $d(d, d2)$ are as follows.

$$\begin{aligned} d(c, c1) &= |xc_{2,i,j} - x_{2,i,j-1}| + |yc_{1,i,j} - y_{1,i,j-1}| \\ d(c, c2) &= |xc_{2,i,j} - x_{2,k,j-1}| + |yc_{1,i,j} - y_{1,k,j-1}| \\ d(d, d1) &= |xc_{1,i,j} - x_{1,i,j-1}| + |yc_{2,i,j} - y_{2,i,j-1}| \\ d(d, d2) &= |xc_{1,i,j} - x_{1,k,j-1}| + |yc_{2,i,j} - y_{2,k,j-1}| \end{aligned} \quad (6)$$

3) The type of tracking frame fusion is shown in Figure 4a.

If $d(a, a1) < d(a, a2)$ and $d(b, b1) > d(b, b2)$, then it is necessary to update the tracking frame, otherwise there is no need to update the tracking frame. The steps to update the tracking box are as follows.

Step 1: Obtain the maximum value of the horizontal and vertical coordinate offset in the two adjacent frames in the continuous N frames of image.

$$\begin{aligned} \Delta x_{2,i,j} &= \max(|x_{2,i,j-1} - x_{2,i,j-2}|, \dots, |x_{2,i,j-N+1} - x_{2,i,j-N}|) \\ \Delta y_{2,i,j} &= \max(|y_{2,i,j-1} - y_{2,i,j-2}|, \dots, |y_{2,i,j-N+1} - y_{2,i,j-N}|) \\ \Delta x_{1,k,j} &= \max(|x_{1,k,j-1} - x_{1,k,j-2}|, \dots, |x_{1,k,j-N+1} - x_{1,k,j-N}|) \\ \Delta y_{1,k,j} &= \max(|y_{1,k,j-1} - y_{1,k,j-2}|, \dots, |y_{1,k,j-N+1} - y_{1,k,j-N}|) \end{aligned} \quad (7)$$

Step 2: Update tracking box.

The first tracking box:

$$\begin{aligned} x_{1,i,j} &= xc_{1,i,j} \\ y_{1,i,j} &= yc_{1,i,j} \\ x_{2,i,j} &= x_{2,i,j-1} + \Delta x_{2,i,j} \\ y_{2,i,j} &= y_{2,i,j-1} + \Delta y_{2,i,j} \end{aligned} \quad (8)$$

The second tracking box:

$$\begin{aligned} x_{1,k,j} &= x_{1,k,j-1} - \Delta x_{1,k,j} \\ y_{1,k,j} &= y_{1,k,j-1} - \Delta y_{1,k,j} \\ x_{2,k,j} &= xc_{2,i,j} \\ y_{2,k,j} &= yc_{2,i,j} \end{aligned} \quad (9)$$

4) The type of tracking frame fusion is shown in Figure 4A. If $d(a, a1) < d(a, a2)$ and $d(b, b1) > d(b, b2)$, then it is necessary to update the tracking frame, otherwise there is no need to update the tracking frame. The steps to update the tracking box are as follows.

Step 1: Obtain the maximum value of the horizontal and vertical coordinate offset in the two adjacent frames in the continuous N frames of image.

$$\begin{aligned} \Delta x_{1,i,j} &= \max(|x_{1,i,j-1} - x_{1,i,j-2}|, \dots, |x_{1,i,j-N+1} - x_{1,i,j-N}|) \\ \Delta y_{2,i,j} &= \max(|y_{2,i,j-1} - y_{2,i,j-2}|, \dots, |y_{2,i,j-N+1} - y_{2,i,j-N}|) \\ \Delta x_{2,k,j} &= \max(|x_{2,k,j-1} - x_{2,k,j-2}|, \dots, |x_{2,k,j-N+1} - x_{2,k,j-N}|) \\ \Delta y_{1,k,j} &= \max(|y_{1,k,j-1} - y_{1,k,j-2}|, \dots, |y_{1,k,j-N+1} - y_{1,k,j-N}|) \end{aligned} \quad (10)$$

Step 2: Update tracking box.

The first tracking box:

$$\begin{aligned} x_{1,i,j} &= x_{1,i,j-1} - \Delta x_{1,i,j} \\ y_{1,i,j} &= yc_{1,i,j} \\ x_{2,i,j} &= xc_{2,i,j} \\ y_{2,i,j} &= y_{2,i,j-1} + \Delta y_{2,i,j} \end{aligned} \quad (11)$$

The second tracking box:

$$\begin{aligned} x_{1,k,j} &= xc_{1,i,j} \\ y_{1,k,j} &= y_{1,k,j-1} - \Delta y_{1,k,j} \\ x_{2,k,j} &= x_{2,k,j-1} + \Delta x_{2,k,j} \\ y_{2,k,j} &= yc_{2,i,j} \end{aligned} \quad (12)$$

3 Results and discussion

In order to evaluate the tracking performance of the proposed algorithm, this research uses the deep learning framework based on the TensorFlow^[26]. The experimental environment uses computer configuration for the 8 GB memory, and CPU is Intel(R) Core (TM) i7-6700 in Ubuntu16.04 operating system. The experiment consists of two parts: training features and tracking targets to obtain the tracking result. Due to the off light state of the pigsty at night, there is no light exposure pigsty, and pigs are sleep at night, so it is not within the scope of this paper.

3.1 Tracking results and analysis of pigs

In order to verify the effectiveness of the proposed algorithm, the designed experiments include target tracking of multi pigs under mutual occlusion. The performance of the tracking algorithm in this study is analyzed by qualitative and quantitative methods.

3.1.1 Tracking results analysis

In the experiment, Faster R-CNN is compared with this algorithm on the tracking effect of the target pigs. Selects 3 test pig video samples to use a trained Convolutional Neural Network for continuous tracking. Figure 5a shows the tracking effect of the Faster R-CNN, and Figure 5b shows the tracking effect of the RP Faster R-CNN.

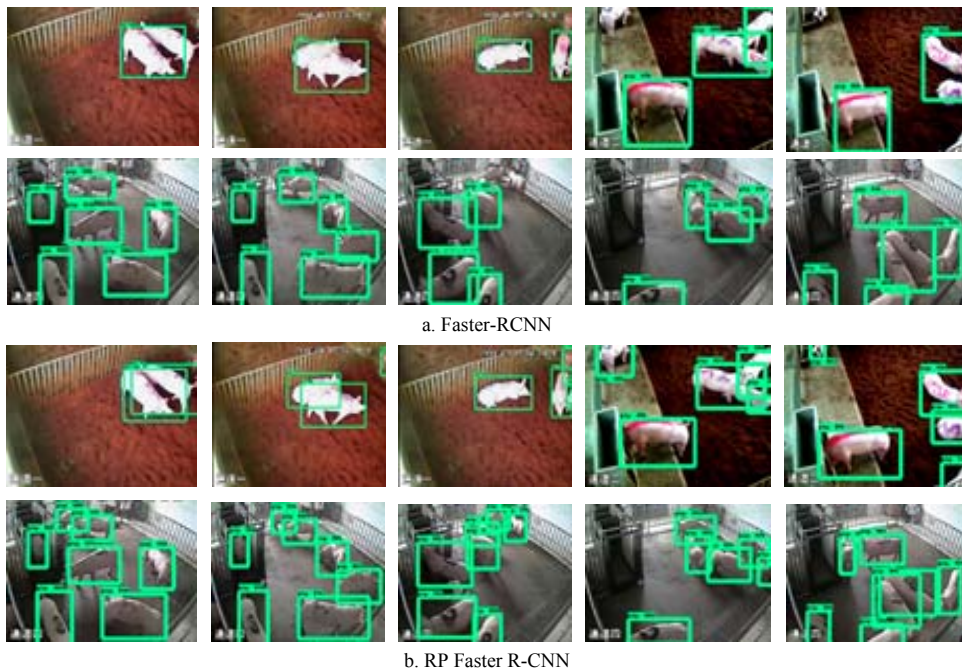


Figure 5 Tracking effect of Faster R-CNN and RP Faster R-CNN

It can be seen by comparing the effects of Figures 5a and 5b: first, the RP Faster R-CNN proposed in this paper can solve the problem of tracking frame fusion caused by the close proximity of targets in traditional Faster R-CNN tracking; second, The RP Faster R-CNN presented in this paper is better than the Faster R-CNN in resolving the effect of missing hits on the target hogs during tracking. This method based on Faster R-CNN automatically extracts the target features, and performs well in solving the multi objective occlusion problem. It shows the feasibility of using RP Faster R-CNN to solve the target tracking problem, and has good robustness and accuracy to the changes of moving objects.

3.1.2 Tracking performance analysis before and after correction

In order to quantitatively analyze the tracking performance of RP Faster-RCNN under different conditions, 2 objective indexes are used to evaluate the tracking effect^[27]: the success rate and the center point error.

1) Success rate: The metric tracking algorithm tracks the target at some point.

$$H_u = \frac{R_g \cap R_t}{R_g \cup R_t}, (u=1, \dots, U) \tag{13}$$

$$H_{avg} = \frac{\sum_{u=1}^U H_u}{U} \tag{14}$$

From the equations above, R_g is the target box area; R_t is the real target box area; U is the total number of target pigs tracked; H_u is the success rate of the u^{th} target pig; H_{avg} is the average success rate of the target pig in the video frame. Normally, if the success rate is less than a preset threshold value of 0.5, then the tracking fails at that moment. If $H > 0.5$, it is considered that tracking success.

2) Center point error: It indicates the Euclidean distance between the center point of the target box tracked and the real center point of the target box. It measured the accuracy of the tracking algorithm on the target tracking. The smaller the value is, the better the tracking effect. In the ideal case, the center point error is 0.

$$err_u = \sqrt{(h_e - h_f)^2 + (y_e - y_f)^2}, (u=1, \dots, U) \tag{15}$$

$$err_{avg} = \frac{\sum_{u=1}^U err_u}{U} \tag{16}$$

In the equations above, (h_e, y_e) represents the coordinates of the center point of the tracking target box area; (h_f, y_f) represents the coordinates of the center point of the real target box area; err_u is the center point error of the u^{th} target pig; err_{avg} is the average center point error of the target pig in the video frame.

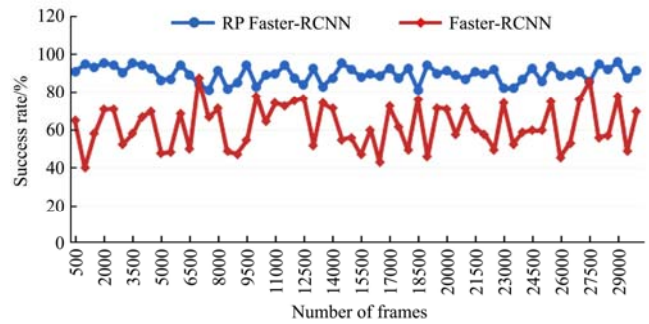
The tracking results obtained by Faster-RCNN and RP Faster-RCNN are analyzed quantitatively, and the real target box area is drawn by hand. To ensure the reliability of the experiment, ten contrast tests were performed on each video during the experiment. Table 1 records the success rate and center point error of the 28000th frame in ten tests of a video. Figure 6 shows a comparison of the success rate and center point error of a typical 1 segment test video sample. The success rate and the center point error are the average of ten tests in this video.

As can be seen from Figure 6, the success rate of RP Faster-RCNN ranges from 80% to 97% in the test, which is generally higher than Faster-RCNN. The overall success rate increased by 44.33%. This means that the method has a lower missed rate than Faster-RCNN. The average center point error of

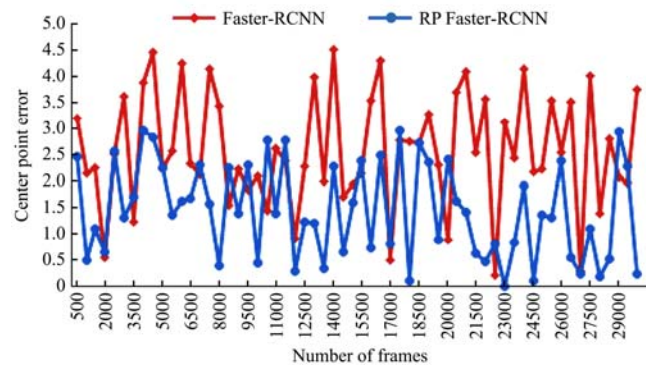
this algorithm is 1.46, so the tracking effect is more stable. The average center point error of Faster-RCNN is 2.60, and the error data fluctuates greatly. RP Faster-RCNN is lower than Faster-RCNN by 43.85%. It shows that RP Faster-RCNN uses the tracking box position correction to reduce the interference caused by the change of the target. It realizes the accurate tracking of target pigs and enhances the robustness of the algorithm, so it improves the success rate and accuracy of tracking.

Table 1 Success rate and center point error of the 28000th frame in ten trials of a video (%)

Number	Faster-RCNN		RP Faster-RCNN	
	Success rate	Center point error	Success rate	Center point error
1	66.13	2.53	91.53	0.82
2	61.57	3.49	89.62	0.20
3	57.81	4.10	92.45	1.01
4	49.46	2.74	90.83	0.64
5	63.50	3.37	88.01	0.49
6	54.30	1.21	92.81	0.00
7	59.52	3.13	89.64	0.25
8	59.84	2.43	93.70	0.85
9	70.97	3.27	90.61	1.83
10	47.38	2.11	92.42	0.11
Average value	59.05	2.84	91.16	0.62



a. Success rate chart



b. Center point error chart

Figure 6 Success rate chart and the central point error chart of the test videos sequence

3.2 Tracking accuracy analysis of multi target pigs

To analyze this algorithm can achieve multi target pigs tracking, and design and calculate the number of pigs in each frame tracked at 4, 5, 6, 7 and 8, respectively.

The multi target tracking results are shown in Figure 7, and the accuracy results are listed in Table 2.

It can be seen from Table 2 that the average accuracy of this algorithm is 83.781%, which can better realize the multi target pigs tracking. It effectively solves the problems of missing and wrong tracking, and it has better robustness.

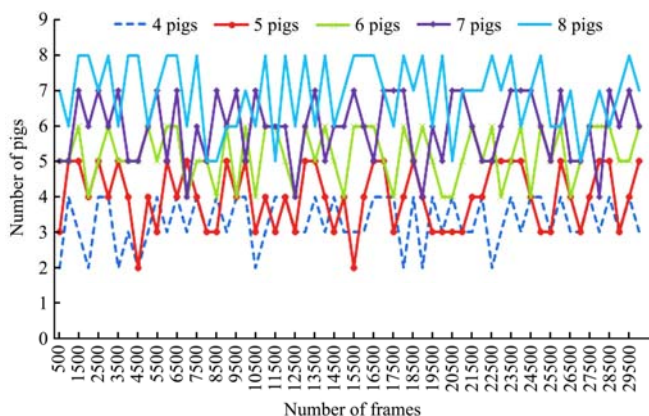


Figure 7 Multi target pigs tracking results of the video sequence

Table 2 Multi target pigs accuracy results of the video sequence (%)

4 pigs	5 pigs	6 pigs	7 pigs	8 pigs	Average accuracy
82.917	80.000	86.667	84.281	85.042	83.781

4 Conclusions

In order to solve the problem that target tracking frames are lost during the visual tracking of pigs, this paper presents a RP Faster-RCNN method to obtain the tracking box of pig based on Faster R-CNN, and corrects the tracking box position offset in consecutive multi frames. To solve the problem of tracking target fusion and target tracking loss due to mutual occlusion, the main conclusions are as follows:

(1) As can be seen from the experimental data, the success rate range and the average center point error of RP Faster-RCNN in the data set are 80%-97% and 1.46 respectively. In term of Faster-RCNN, the success rate range and average center point error are 40%-85% and 2.60. Thus, RP Faster-RCNN is generally superior to Faster-RCNN, which overcomes the complex background disturbances of target pig video tracking. The accuracy of multi target pigs tracking is 83.78%, so it can accurately track multi pigs when the targets occlude each other and have good robustness.

(2) The RP Faster-RCNN proposed is applied to pig tracking, the tracking effect and the accuracy of detecting the target box are obviously superior to the Faster-RCNN. The application of deep learning to target tracking technology improves the tracking precision and enhances the practicability of the system. It provides a new method to solve the problem of target tracking, which is of great significance for improving the target occlusion in complex scenes.

Acknowledgments

The authors acknowledge that this research was financially supported by the National High Technology Research and Development Program of China (2013AA102306).

[References]

[1] Li Y Y, Sun L Q, Zou Y B, Li Y. Individual pig object detection algorithm based on Gaussian mixture model. *Int J Agric & Biol Eng*, 2017; 10(5): 186–193.
 [2] Yu S. Tracking algorithm based on multi-feature detection and target association of pigs on large-scale pig farms. *Journal of Information & Computational Science*, 2015; 12(10): 3837–3844.
 [3] Sun L, Li Z, Duan Q, Sun X, Li J. Automatic monitoring of pig

excretory behavior based on motion feature. *Sensor Letters*, 2014; 12(3): 673–677.
 [4] Porto S M C, Arcidiacono C, Anguza U, Cascone G. A computer vision-based system for the automatic detection of lying behavior of dairy cows in free-stall barns. *Biosystems Engineering*, 2013; 115(2):184–194.
 [5] Zuo S, Jin L, Chung Y, Park D. An index algorithm for tracking pigs in pigsty. *International Conference on Industrial Electronics and Engineering*, 2015; pp.797–804.
 [6] Sanchez-Matilla R, Poiesi F, Cavallaro A. Online multi-target tracking with strong and weak detections. *European Conference on Computer Vision*, 2016; pp.84–99.
 [7] Kamal A T, Bappy J, Farrell J, Roychowdhury A. Distributed multi-target tracking and data association in vision networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2016; 38(7): 1397–1410.
 [8] Lecun Y, Bengio Y, Hinton G. Deep learning. *Nature*, 2015; 521(7553): 436.
 [9] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. *International Conference on Neural Information Processing Systems*. Curran Associates Inc. 2012; pp.1097–1105.
 [10] Sun Y, Wang X, Tang X. Deeply learned face representations are sparse, selective, and robust. *Computer Vision and Pattern Recognition*. IEEE, 2015; pp.2892–2900.
 [11] Zhou Y C, Xu T Y, Zheng W, Deng H B. Classification and recognition approaches of tomato main organs based on DCNN. *Transactions of the CSAE*, 2017; 33(15): 219–226. (in Chinese)
 [12] Ma C, Huang J B, Yang X, Yang M H. Hierarchical convolutional features for visual tracking. *IEEE International Conference on Computer Vision*. IEEE, 2015; pp.3074–3082.
 [13] Andriyenko A, Roth S, Schindler K. An analytical formulation of global occlusion reasoning for multi-target tracking. *IEEE International Conference on Computer Vision Workshops*, 2011; pp.1839–1846.
 [14] Dong X, Shen J, Yu D, Wang W, Liu J, Huang H. Occlusion-Aware Real-Time Object Tracking. *IEEE Transactions on Multimedia*, 2017; 19(4):763–771.
 [15] Hua Y, Alahari K, Schmid C. Occlusion and motion reasoning for long-term tracking. *Computer Vision – ECCV 2014*. Springer International Publishing, 2014; pp.172–187.
 [16] Yang H, Alahari K, Schmid C. Online object tracking with proposal selection. *IEEE International Conference on Computer Vision*. IEEE, 2015; pp.3092–3100.
 [17] Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. *Tech Report (v5)*, UC Berkeley, 2013; pp.580–587.
 [18] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans Pattern Anal Mach Intell*, 2015; 39(6): 91–99.
 [19] Uijlings J R, Sande K E, Gevers T, Smeulders A W M. Selective search for object recognition. *International Journal of Computer Vision*, 2013; 104(2): 154–171.
 [20] He W, Yamashita T, Lu H, Lao S. SURF Tracking. *IEEE International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October*. DBLP, 2009; pp.1586–1592.
 [21] Takita A, Ogawa S, Saji H. Vehicle motion tracking from infrared images using background subtraction method. *16th ITS World Congress and Exhibition on Intelligent Transport Systems and Services*, 2009.
 [22] Han S, Mao H, Dally W J. Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. *Fiber*, 2015; 56(4): 3–7.
 [23] Bay H, Ess A, Tuytelaars T, Gool L V. Speeded-up robust features (SURF). *Computer Vision & Image Understanding*, 2008; 110(3): 346–359.
 [24] Lowe D G. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004; 60(2): 91–110.
 [25] Niedermeier R, Sanders P. On the manhattan-distance between points on space-filling mesh-indexings. *Technical Report iratr-1996-18*, Universität Karlsruhe, Informatik für Ingenieure und Naturwissenschaftler, 1996.
 [26] Abadi M. TensorFlow: Learning functions at scale. *Acm Sigplan Notices*, 2016; 51(9): 1–1.
 [27] Wu Y, Lim J, Yang M H. Online object tracking: A benchmark. *IEEE Conference on Computer Vision and Pattern Recognition*, 2013; pp.2411–2418.