Feed weight estimation model for health monitoring of meat rabbits based on deep learning

Enze Duan, Liangju Wang, Hongying Wang*, Hongyun Hao, Rangling Li

(*College of Engineering, China Agricultural University, Beijing 100083, China*)

Abstract: With the development of precision livestock farming, non-contact health monitoring technology is particularly important in the breeding process.　To help improve the management of the rabbit breeding industry, a remaining feed weight (RFW) estimation model was developed based on the image segmentation method.　The model proposed in this study consisted of a feed instance segmentation neural network and feed weight estimation network.　Feed instance segmentation neural network was based on the improved Mask Region-based Convolution Neural Network (Mask RCNN), the state-of-art image segmentation method, and the PointRend algorithm was used to replace the original network head.　Through an adaptive subdivision strategy, the boundary points were segmented with fine details.　Features were extracted from the segmentation results and used as the input of the feed weight estimation network based on the Back Propagation (BP) algorithm. The model was applied in rabbit breeding to explore the relationship between RFW and the mortality probability of meat rabbits. The model evaluation results showed that the Average Precision (AP) value of the feed instance segmentation neural network was 0.987, the Mean Pixel Accuracy (MPA) value was 0.985.　The correlation coefficient of the feed weight estimation network was 0.97, the Mean Squared Error (MSE) was 208.3, and the Mean Absolute Error (MAE) was 10.51 g.　The practical application results showed that the feed intake of the unhealthy meat rabbits would decrease significantly.　When the RFW was more than 50% of feed quantity, the mortality probability of the rabbit was more than 85%; when the RFW was more than 65% of feed quantity, all the rabbits finally died in a short time.　Therefore, there is a significant correlation between RFW and the mortality probability of rabbits, by which this proposed model can help farms to monitor the health of meat rabbits by predicting RFW.

Keywords: meat rabbit, remaining feed, weight estimation, convolutional neural network, deep learning, health monitoring

DOI: 10.25165/j.ijabe.20221501.6797

1　Introduction

As a traditional special economic animal breeding industry, China is committed to the mechanization and intelligence of the rabbit industry.　Health monitoring regularly is one of the most important and heaviest tasks in rabbit breeding farms because dead or dying rabbits may cause the spread of the diseases.　However, the research of rabbit epidemic has invested a lot of energy in rabbit population tracking and pathological anatomy, but there are few reports about disease prevention in the process of rabbit breeding[1-3].

Large-scale rabbit breeding mostly blocks the spread of the disease through the batch and standardization of production management.　However, the disease monitoring of an individual meat rabbit still depends on the regular inspection of rabbit houses by breeders.　In actual rabbit breeding, the diseases monitoring by

breeders would be costly and consuming.　On the other hand, the activity of rabbits in cages is significantly lower than that in the wild, so most rabbit diseases do not have any symptoms in the early stage[4], which reduces the accuracy of manual inspection. Besides, to ensure meat quality and reduce drug resistance of farm animals, the addition of veterinary drugs and antibiotics to feed is gradually restricted in many countries and districts, which makes rabbits more susceptible to infection[5,6].

At present, most intelligent health monitoring technologies for farm animals are based on sensors and machine vision to extract images, sounds, temperature, weights, and other features.　Cuan et al.[7] proposed a chicken sound convolutional neural network to detect chickens with avian influenza.　Four features of the chicken sounds were extracted and combined into feature maps.　The proposed network recognized the sounds of healthy chickens and chickens with avian influenza.　Zhang et al.[8] used the thermal infrared images of dairy cow eyes and udders to detect dairy cow mastitis, the temperature difference between the eyes and the udders was obtained based on the YOLOv3 algorithm.　Zhao et al.[9] designed a real-time video capturing system for dairy cows, and the breathing rate was monitored based on the optical flow algorithm.　Cows breathing abnormity was detected according to the duration of each breath.　Hertem et al.[10] detected lameness of cows with multivariate continuous sensing.　The sensors measured neck activity and ruminating time of healthy cows and lame cows, and the dataset was used as input of the logistic regression model. Combining the advanced sensors and network analysis methods has become a valuable tool in farm animal sciences as the whole system can easily extract features that are difficult for humans to

distinguish.

Most of the current researches on intelligent disease detection and prevention technology of farm animals mainly focus on large animals such as pigs, cattle, and sheep which is due to the breeding scale and consumption market of these farm animals being much larger than that of rabbits. Rabbit farming is predominantly a closed cycle performed on a single farm in individual cages[11]. For each cage, the activity space of the rabbits is crowded, and the number of breeding rabbits is uncertain. It is complicated to monitor the health of rabbits directly: the cost of the wearable sensor is too high, and the sound features are hard to obtain because rabbits seldom bark, and cages and equipment block the view of the machine. Therefore, in large-scale rabbit breeding, the studies on intelligent disease detection and prevention technology, especially on the prediction of rabbit mortality probability are few.

To a certain extent, the remaining feed weight (RFW) can reflect the health of rabbits. All-in-all-out production is widely used in rabbit breeding industry, and it is obvious that the feed intake of rabbits in the same batch is close[12]. According to the previous studies[4,13-16], most of the common rabbit diseases, especially gastrointestinal diseases, respiratory diseases, and parasitic diseases, set off the loss of appetite and reduce the average daily feed intake (ADFI) of rabbits. Therefore, the more the RFW is, the higher the mortality probability. Because of the habit of rodents, using sensors or other engineering means to automatically weigh the feed for each cage is expensive and unsafe. However, it was convenient to obtain the feed images of the feed boxes, so an RFW estimation model with feed images was developed. The model consisted of a feed instance segmentation neural network and a feed weight estimation network. The feature of feed instances was extracted based on feed instance segmentation neural network, which employed the Mask Region-based Convolution Neural Network (Mask RCNN) framework and used the PointRend algorithm as the network head to extract boundary. The feed weight estimation network was based on Back Propagation Neural Network (BPNN) and used the extracted features as input. With the proposed RFW estimation model, the relationship between RFW and the mortality probability of the rabbit had been explored in the breeding farms to prove the practicability and feasibility of the model.

2   RFW estimation model construction principles

The overall flow of the proposed method was to extract the target features from feed images with a convolution neural network, then input the features to a BP neural network as input to estimate the RFW. According to the output of the trained BP neural network, the relationship between RFW and the mortality probability was verified by experiments. Thus this study established the relationship between feed images and prediction of the rabbit death

2.1   Experiment device

In order to calculate RFW by feed image, the shape of the feed box was analyzed to determine the image capturing method first. In this study, the model is designed for the most commonly used feed box in China's standardized meat rabbit farm. Other types of feed boxes can refer to this article for similar analysis. Figure 1 shows the structure of the feed box. The whole feed box consists of a feeding inlet and a feeding trough. The feeding inlet, where feed was put in, was similar to the Mitsubishi column and located outside the rabbit cage. The feeding trough, where meat rabbits eat feed, was similar to the cuboid and located inside of the rabbit cage. Due to the irregular structure of the feed box, the remaining feed (RF) in the feed box may exist both at the feeding inlet and feeding trough.
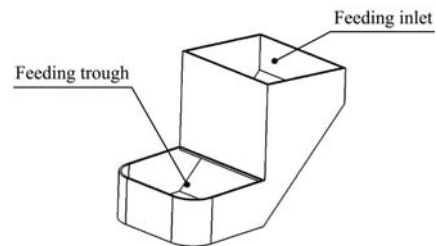

Figure 1   Structure of the feed box

2.2   Feed instance segmentation network

2.2.1   Architecture of Mask RCNN

Based on the Faster RCNN[17], Mask RCNN added a segmentation mask generating branch and used the RoIAlign algorithm instead of the traditional RoIPooling algorithm to preserve the accuracy of the spatial positions of targets[18]. As shown in Figure 2, feed images were used as the input of Mask RCNN, and Mask RCNN outputted the feed instance.
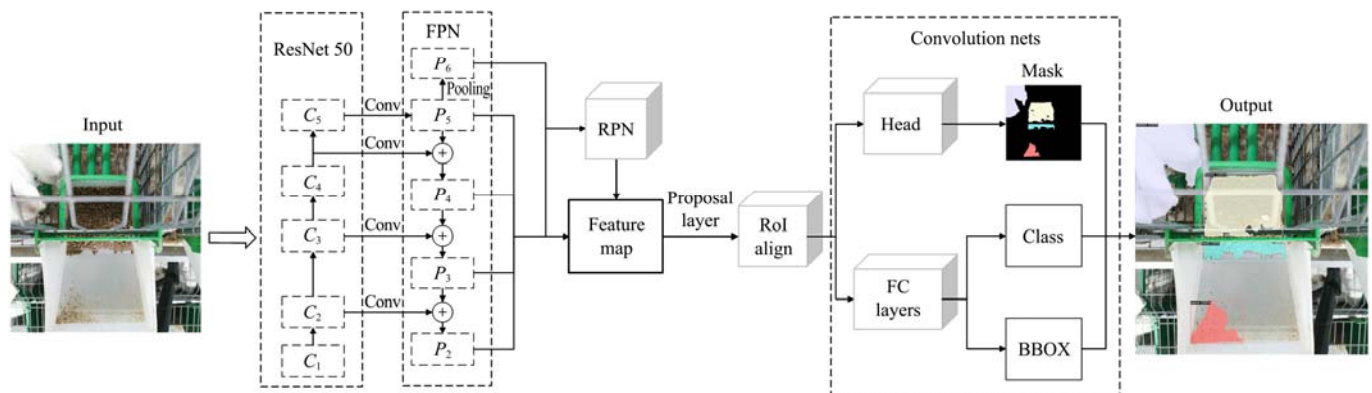
Note: ResNet: Residual Network; Conv: Convolution; FPN: Feature Pyramid Network; RPN: Region Proposal Network; RoI: Region of Interest; FC Layers: Fully Connected Layers; BBOX: Bounding Box

Figure 2   Structure of feed segmentation network based on Mask RCNN

The backbone network provides inputs for Feature Pyramid Network (FPN). Different weight layers make up the backbone network. For complex features, deeper networks may result in higher accuracy but the training time of the model will be longer.

In this study, because the features to be extracted were not complex, FPN based on ResNet-50 was used as the backbone network to alleviate the difficulties of gradient disappearance and training degradation[19].

ResNet-50 consists of five stages, corresponding to five different scales of feature map as $[C_1, C_2, C_3, C_4, C_5]$. Through a bottom-up structure, the stride of these five feature maps gradually doubled, and are used to establish the feature pyramid of FPN. Specifically, C_1 is not used because the large size takes a huge time to calculate a new feature map. The whole process is shown as follows:

$$\begin{cases} P_i' = \text{sum}(U_{i+1}, \text{conv}(C_i)) \\ U_i = \text{upsample}(P_i') \\ P_i = \text{conv}(P_i') \end{cases} \tag{1}$$

$$P_6 = pooling(P_5)$$

where, U_i represents the feature map obtained by upsampling; C_i represents the i-th feature map output at each stage; P_i represents the i-th new feature map; P_i' represents the feature map obtained by the element-by-element alignment operate; conv represents the convolution operation; sum represents the element-by-element alignment operation; upsample represents the upsampling operation; pooling represents the pooling operation which stride is 2.

For new feature maps $[P_2, P_3, P_4, P_5, P_6]$, the Region Proposal Network (RPN) uses a sliding window to scan them and find the regions of interest (RoI) areas where the feed exists. The first layer of RPN outputs the categories and boundary coordinates of the areas, and judges whether these regions are foreground or background. For the foreground, the second layer of RPN uses a different size of anchors to correct the bounding box of the target. The Non-Maximum Suppression (NMS) is used to remove overlapping anchors, and anchors with higher foreground scores are sent to the next stage.

RoIAlign aligns the extracted features with the original region proposal. Bilinear interpolation is used in RoIAlign to compute the exact values of the input features, then the size of the anchor is adjusted to a fixed size. The obtained features are fed to the Fully Connected (FC) layer and the segmentation convolution layer.

The loss function of Mask RCNN is used to train bounding box regression, classification, and mask prediction branches, which can be described as follows:

$$L = L_{cls} + L_{box} + L_{mask} \tag{2}$$

$$L_{cls} = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) \tag{3}$$

$$L_{box} = \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \tag{4}$$

$$L_{mask} = -\frac{1}{S_m} \sum_i \left[m_i^* \lg p(m_i) - (1 - m_i^*) \lg(1 - p(m_i)) \right] \tag{5}$$

where, L_{cls} represents the classification error; L_{box} represents the bounding box regression error; L_{mask} represents the mask error; N_{cls} represents the number of categories; $L_{cls}(p_i, p_i^*)$ represents the logarithmic loss of RoIs; p_i represents the probability that the anchor is predicted to be positive samples; p_i^* represents the probability that the anchor is predicted to be negative samples; N_{reg} is the pixel number in the feature map, $L_{reg}(\cdot)$ is a smooth function; t_i is the predicted offset of anchors in RPN; t_i^* is the true offset of anchors in RPN; S_m is the size of the mask; m_i^* is the label value of mask; $p(m_i)$ is the probability that m_i^* is predicted to be right label value.

2.2.2 Optimized boundary feature based on PointRend technique

Image segmentation tasks of Mask RCNN focus on regular grids of the input feed image and make excessive predictions over all points on the output grid. Paying less attention to boundary segmentation smoothness increases the loss of prediction, which makes the boundary of the feed mask not clear. To solve this problem, PointRend uses a subdivision strategy to adaptively select a non-uniform set of points at which to compute labels[20]. The modules in the PointRend consist of point selection, point-wise feature extraction, and point head.

The point selection module selects points for predicting segmentation labels flexibly and adaptively and focuses on the points located near high-frequency ports such as object boundaries. For inference, the strategy of point selection is to render the output image in a coarse-to-fine manner. In each iteration, the points on a regular grid will be predicted coarsest first. Based on bilinear interpolation, the low spatial resolution feature map is up-sampled and achieved the desired spatial resolution, as shown in Figure 3. Then on the denser grid, the most uncertain points whose degree of confidence is less than 0.5 (e.g., Confidence Interval) are selected. The points are selected as follows:

$$n_i^* = \arg\min_{n_i} |p(n_i) - 0.5| \tag{6}$$

where, $p(n_i)$ is the probability for point n_i belonging to the binary mask; n_i^* is the selected point.
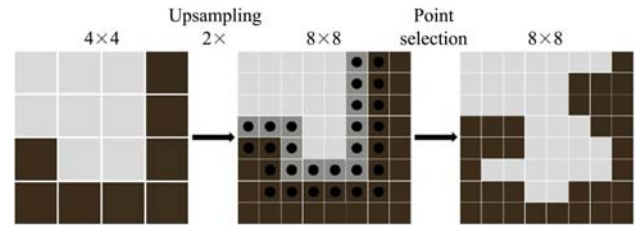
Note: Grey grids and black grids are the pixel points of different labels on the feature map; Black points represent the selected grids on edges; Arrows are the process of upsampling and point selection.

Figure 3   Adaptive subdivision step for one feed image

For training, the point selection strategy is a non-iterative strategy based on random sampling. First, there are kN points ($k>1$) randomly sampled from the feature map. Then these points are sorted according to their uncertainty estimate. The most uncertain βN ($\beta \in [0,1]$) points are selected, these βN points are concentrated in the most uncertain area and their distribution is relatively uniform.

The point-wise feature extraction module is constructed by concatenating fine-grained features and coarse prediction features on selected βN points. The fine-grained feature is computed by bilinear interpolation on the feature maps, which contains the finely detailed segmentations. The coarse predicted feature is a 2D vector with more contextual and semantic information that provides a more globalized context and helps convey the semantic classes. Features can be extracted from a single feature map and multiple feature maps.

The point head module is a Multilayer Perceptron (MLP) with three hidden layers and 256 channels, which predict labels based on the above point-wise features. MLP shares weights across all points and regions. This point head concatenates coarse prediction and fine-grained feature vectors. Rectified Linear Unit (ReLU)[21] is used on the hidden layers of MLP, and the sigmoid activation function is applied to its outputs.

2.3 Parameter extraction method of the feed

The retrieved features parameters must be representative to estimate RFW. As mentioned in Section 2.1, the parameters of feed contour in feeding inlet and feeding trough needed to be extracted separately. Figure 4 shows the segmentation result of

the feed instance segmentation network described in Section 2.2. For a briefer description, the feed at the feeding trough was named 'feed1', and the feed at the feeding inlet was named 'feed2'.

Note: The feed at the feeding trough was named 'feed1' and covered with a yellow mask; the feed at the feeding inlet was named 'feed2' and covered with a blue mask; the boxes were the bounding boxes of the instances; the percentage numbers were the confidence scoring.

Figure 4   Segmentation result of the feed

'Feed1' is covered with a yellow mask, and 'feed2' is covered with a blue mask. Due to the proportional relationship between the feed area and RFW, the area of the feed mask provides good approximations for the weights. For each mask, the number of pixels was calculated with a given threshold in RGB color space.

Let C_i (i = 1, 2) the number of pixels per mask, where C_i is the area of feed masks.

The image was a 2D representation of a 3D object.   In this study, only C_i was used to estimate RFW may cause great errors. Thus the thickness of the feed also needed to be extracted.   In the top view, the thickness of the feed cannot be measured directly. However, based on the basic principle of perspective, the feed box will be filled more with the increase of the feed thickness.   Figure 5 is an enlarged drawing of the feeding trough captured from the whole image, as shown in Figure 5, when the thickness of the feed increased, the distance between the edges of the feed mask and the feed box will be shortened.

Since the length and width of the feed box are fixed, the distance can be expressed as follows:

$$\begin{cases} L_x = W_B - W_F \\ L_y = H_B - H_F \end{cases} \tag{7}$$

where, W_B represents the length of the feed box; H_B represents the width of the feed box; W_F and H_F represent the length and width of feed segmentation, respectively; L_x and L_y represent the distance between edges of the feed segmentation and the feed box in the X-axis and Y-axis, respectively.

As the feed box's size is known, the size of the feed mask can be obtained from the bounding box of the mask.   For each of the images, a feature vector F has the above-mentioned features as its elements as follows:

$$F = (C_1, L_{x1}, L_{y1}, C_2, L_{x2}, L_{y2}) \tag{8}$$

a. Estimation index with thick feed layer   b. Estimation index with thin feed layer

Note: Red lines represent the edges of the feed masks and the feed boxes.   The sum of the distances between the vertical lines represents the distance between the feed and the magazine in the X direction, and the sum of the distances between the horizontal lines represents the distance between the feed and the magazine in the Y direction.

Figure 5   Schematic diagram of the feed thickness estimation index

2.4   Feed weight estimation network

Based on the error BP algorithm, BPNN can store and learn a large number of input and output patterns without determining the mathematical equations to describe the relationship[22].   This study aimed to predict RFW with the feature of 2D images.   Based on the premise that the linear relationship between input and output is distorted, BP neural network is used to fit the data.   The nonlinear mapping and generalization characteristics of the BP neural network are helpful to building an appropriate estimation model[23].

The feature vector F which was obtained in Section 2.2, provided neural units of the input layer of the BPNN, used the weight weighed in advance as neural units of the output layer. Then the number of neural units of the hidden layer can be calculated as follows[24]:

$$\begin{cases} l < n - 1 \\ l < \sqrt{(n+m)} + a \\ l = \log_2 n \end{cases} \tag{9}$$

where, n is the number of neural units of inputting layer; m is the number of neural units of the output layer; l is the number of neural units of the hidden layer; a is a constant range from 1 to 10. Therefore, $l \in [3,12]$.

The structure of BPNN in this study is shown in Figure 6. Due to the linear relationship between the input and output being distorted, there is no nonlinear activation function, such as ReLU or softmax, used in the network[25,26].

3   Materials and methods

3.1   Data acquisition

The feed images and RFW were recorded over several days in August 2020 in a meat rabbit breeding farm in Henan Province, China.   500 randomly selected cages were used to collect images of RFW.   As shown in Figure 7, to acquire as many features of the RF as possible, the camera was placed directly above the junction of the feeding inlet and feeding trough, so the captured image contained both regions at the same time.
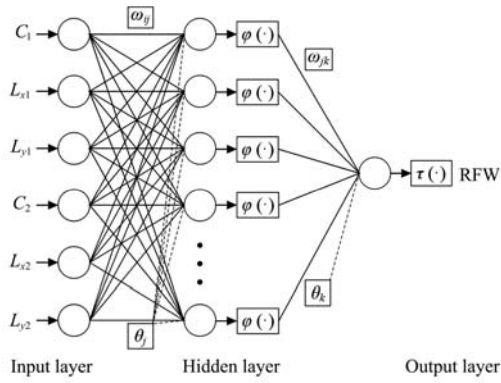
Note: C_1 and C_2 represent the area of each feed mask. L_{x1} and L_{x2} represent the distance between edges of the feed segmentation and the feed box in the X-axis; L_{y1} and L_{y2} represent the distance between edges of the feed segmentation and the feed box in the Y-axis. $\varphi(\cdot)$ and $\tau(\cdot)$ stands for activation function in hidden and output layers; ω_{ij}, ω_{ik}; θ_j, θ_k represent weights and bias value of each layer; RFW represents remaining feed weight.

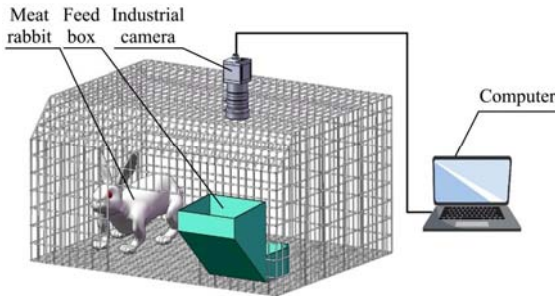Figure 6　Structure of feed weight estimation network

Figure 7　Schematic of the method of image acquisition

When the images were captured, the remaining feed in the feed box was weighed and recorded at the same time. There was not only pellet feed, but also rabbit manure, feed powder, and other impurities in the feed box, so the pellet feed was separated from impurities before weighing. The model of the camera was MV-CA020-10GC (Hikvision, China), the resolution was 1624 pixels × 1240 pixels, the image size was 1920 pixels × 1020 pixels, and the format of images was JPG.

3.2　Algorithm platform

The object detection and segmentation algorithms were implemented with the Detectron2 platform based on PyTorch deep learning framework. The weight prediction algorithm was implemented with MATLAB R2016a. The model was trained on a computer with Intel® Core™ i7-9700K CPU and NVIDIA GeForce RTX 2080 GPU, the system is Ubuntu 20.0.

3.3　Experiment of feed instance segmentation network

The size of obtained images was scaled to 1024×1024 to meet the input of the feed segmentation network. To avoid overfitting, the images were rotated and horizontally flipped to augment the training data. The training dataset contained 500 different images, and the validation set contained 100 different images.

To construct the training dataset, the preprocessed images were annotated by the open-source image annotation tool Labelme. There were feed pellet, mash feed (the broken feed pellet that meat rabbit does not intake), and rabbit feces in the feed box. Also, meat rabbits may interrupt the feed box. In order to avoid interference, five types of categories were annotated, 1) feed pellet at the feeding trough; 2) feed pellet at the feeding inlet; 3) rabbit feces; 4) mash feed; 5) rabbit.

Common Object in Context (COCO) dataset was introduced using transfer learning to solve the problem caused by a small training set before training. COCO is a type of large-scale object detection, segmentation, and captioning dataset which helps to accelerate the training process. ResNet-50 and PointRend were used as the backbone and network head architecture, respectively. The whole training process lasted 30 h with a 0.001 learning rate. In addition, the original Mask RCNN, which uses ResNet-50 as backbone and Faster-RCNN with ResNet as head architecture, was trained to compare with the proposed feed segmentation neural network.

3.4　Experiment of feed weight estimation network

Another dataset was used to train the feed weight estimation network. 500 new feed images were predicted by the trained feed instance segmentation network, and its outputs were sent to the parameter extraction method, then feature vector $\boldsymbol{F}$ was obtained.

Vector $\boldsymbol{F}$ and the RFW were used as training inputs and training output of the BPNN, respectively, and the proportion of training set, validation set, and test set was 70%:15%:15%, the number of neurons in the hidden layer was 4, 5, and 6, respectively. The training function was set as three common functions: Trainlm (Levenberg-Marquardt), Trainbr (Bayesian Regulariza-tion), and Trainscg (Scaled Conjugate Gradient). Adaption learning function and transfer function were set to Learngdm and Tansig according to experience. The whole training process iterated 1000 times with a 0.01 learning rate.

3.5　Performance evaluation methods

Average Precision (AP) and Mean Pixel Accuracy (MPA) were used to evaluate the performance of feed instance segmentation network[27].

The AP compares the ground-truth bounding box to the detected box and returns a score. The higher the score, the more accurate the model is in its detections.

$$P = \frac{TP}{TP + FP} \tag{10}$$

where, P is the precision; TP is the true positive; FP is the false negative.

The MPA reports the percent of pixels in the image which were correctly classified:

$$MPA = \frac{1}{k} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij}} \tag{11}$$

where, k is the total number of categories including the background; p_{ii} is the total number of pixels whose real pixel class is i and predicted as i, and p_{ij} is the total number of the pixel whose real pixel class is i but predicted as j.

To evaluate the performance of the feed weight estimation network based on BP neural network, Correlation Coefficient (r), Mean Squared Error (RMSE), Mean Absolute Error (RMAE) are used, respectively.

$$r = \frac{Cov(y_i, \hat{y}_i)}{\sqrt{Var[y_i]Var[\hat{y}_i]}} \tag{12}$$

$$RMSE = \frac{1}{n}(y_i - \hat{y}_i)^2 \tag{13}$$

$$RMAE = \frac{1}{n} \sum_{i=1}^{n} |(y_i - \hat{y}_i)| \tag{14}$$

where, R is Correlation Coefficient; RMSE is Mean Squared Error; RMAE is Mean Absolute; y_i is the predictive value; $\hat{y}_i$ is the actual value; n is the sample capacity.

3.6　Practical application experiment of the proposed model

This study aimed at monitoring rabbit mortality probability by predicting the RFW with machine vision, so it is necessary to explore the relationship between RFW and the mortality probability

of the rabbits.

1273 rabbits that fed in 694 cages were chosen. Their RFW was estimated, and their death was recorded in this experiment. Those rabbits were judged in two age categories: 36-75 days old and 240 days old, and their Average Daily Feed Intake (ADFI) was restricted[28]. Rabbits were divided into four groups according to their age at the beginning of the experiment.

There were two rabbits in each cage for 36 to 75 days old rabbits. For 240 days old rabbits, there was one rabbit in each cage. Before daily feeding (1:00 p.m.), the cages which had RF would be marked, and the feed images of these cages were captured. Then the pellet feed was separated, weighed, and recorded next. After that, the feed box was emptied, all the rabbits would be fed by standard and recorded the health condition.

During the experiment, the experiment would be repeated until the rabbit dead, or there was no remaining feed in the feed box, which proved that the rabbits were healthy. At the same time, the proposed model was used to estimate the weight of the feed in the images.

4   Results and discussion

4.1   Results of feed instance segmentation network

With the same iterations and the same training dataset, the comparison of training accuracy and total loss of the proposed feed segmentation neural network and the original Mask RCNN is shown in Figure 8.

a. Comparison results of training accuracy

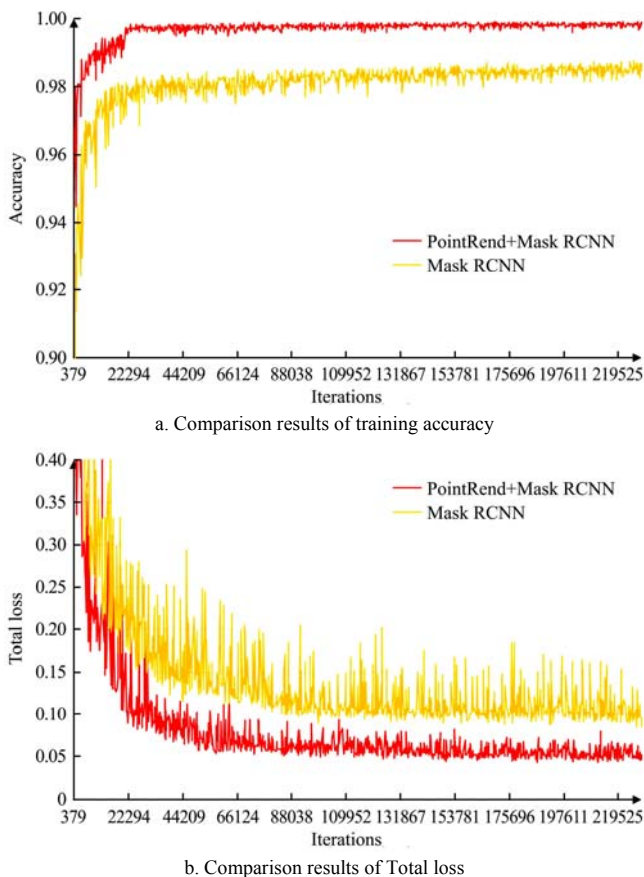b. Comparison results of Total loss

Figure 8   Comparison results of Mask RCNN and PointRend

According to Figure 8, the accuracy of the proposed feed instance segmentation network converges at 0.997, while the accuracy of the original Mask RCNN converges at 0.985. The total loss of the proposed feed segmentation network converges at 0.046, while the total loss of the original Mask RCNN converges at 0.073. The results showed that compared with the original Mask

RCNN, the proposed feed segmentation network has higher segmentation accuracy and lower total loss during training.

For the same image, the segmentation results of the two networks are shown in Figure 9, the segmentation results of instance contour and scattered feed pellet are shown, respectively.
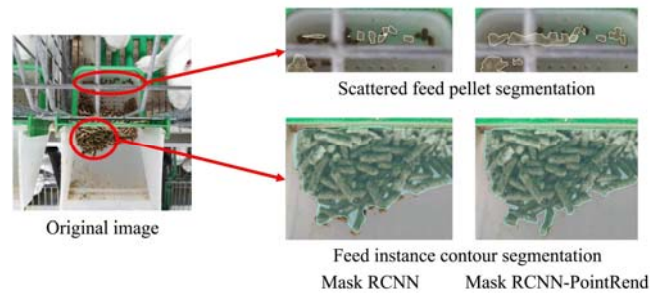

Figure 9   Segmentation results of two networks for the same image

Compared with the ground truth in Figure 9, it could find that both of the networks had extracted the feed mask from the original image, but the proposed Mask RCNN-PointRend model performed better than the Mask RCNN. For the scattered feed pellet, the Mask RCNN recognized fewer feed instances than the proposed model. For the feed instance contour segmentation, the boundaries of the feed pellet by the proposed model appeared to be much clearer and smoother than the Mask RCNN. The edge and corners of the feed in the mask could match correctly with the original image. Figure 10 shows other results of segmentation by the proposed network. It was clear to find that the network could accurately segment the mesh feed or the hollow part of the feed regions.

Due to the fluidity of the feed pellet, hollow holes may appear in the center of the feed area, where no feed exists. These hollow holes are highlighted as shown in Figure 10b.

a. Mash feed instance segmentation

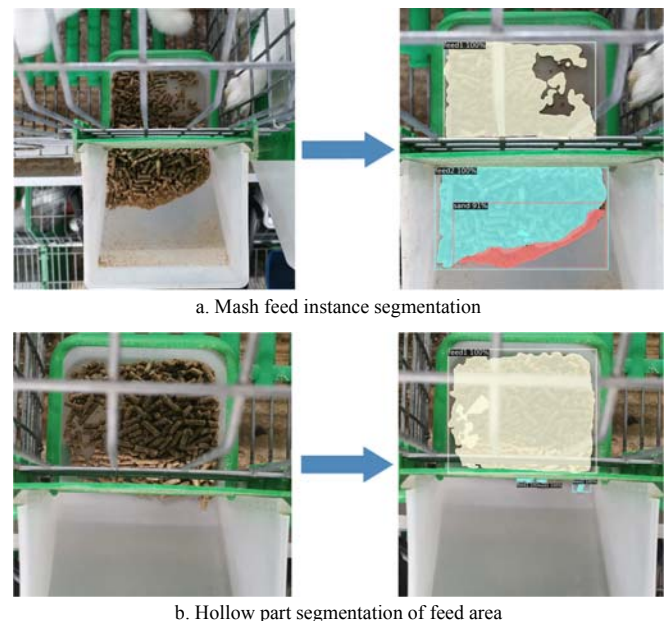b. Hollow part segmentation of feed area

Figure 10   Other segmentation results by the proposed model

The performance evaluation method described in Section 3.4 was used in the same validation set to evaluate these two networks. The results showed that the AP value of the proposed method, which represents the accuracy of object detection, was up to 0.987, which was higher than the 0.940 of the Mask RCNN algorithm. The MPA value of the proposed method, which represents the accuracy of instance segmentation, was up to 0.985, which is

higher than the 0.930 of the Mask RCNN algorithm.   The average running time on an image of the proposed method was 0.23 s, which was lower than the 0.37 s of the Mask RCNN.   In conclusion, the accuracy of object detection, the accuracy of instance segmentation, and the execution speed of the algorithm had increased 4%, 5%, and 37%, respectively.

4.2   Results of feed weight estimation network

The feed weight estimation network has been trained with the parameter that the hidden layer neurons number was 4, 5, and 6, the training function was Trainlm, Trainbr, and Trainscg.   The performance evaluation results are shown in Table 1.

Table 1   Evaluation results of feed weight estimation network

Model type	r	RMSE	RMAE
4-Trainlm	0.889	758.66	18.25
4-Trainbr	0.964	255.43	12.24
4-Trainscg	0.941	418.17	14.87
5-Trainlm	0.965	263.07	11.89
5-Trainbr	0.968	229.80	10.89
5-Trainscg	0.939	433.21	15.17
6-Trainlm	0.930	456.37	15.94
6-Trainbr	0.971	208.33	10.51
6-Trainscg	0.926	522.92	16.15

According to Table 1, the performance of the feed estimation network was the best when the number of hidden layer neurons was 6, and the training function was Bayesian Regularization.   The Correlation Coefficient of the network was 0.971, RMSE was 208.33 and RMAE was 10.51 which was the group's minimum.   The result proved that the proposed feed estimation network could estimate the feed weight with higher precision, and the predicted weight of feed was close to the true weight.

4.3   Results of practical application experiment

4.3.1   Predict results of the proposed model

The proposed model was used in all the cages.   There were 86 cages detected that had remaining feeds, and all the empty feed boxes were recognized.   By inspection with artificial observation, the proposed model could 100% detect whether RF existed in the feed boxes.

The relationship between the predicted weight and the true weight of the 89 cages is shown in Figure 11.

$$y = 0.9796x + 3.8595$$
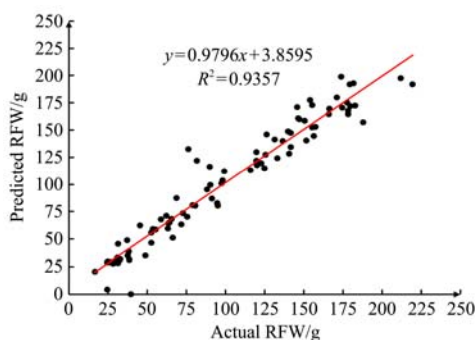$$R^2 = 0.9357$$

Figure 11   Prediction results of RFW estimation model

According to Figure 11, the relationship between the predicted weight and the true weight was linear for the proposed model, the r is 0.97.

4.3.2   Predict results of the proposed model

For all 1273 rabbits in this experiment, the number of deaths was 52, while the number of the cages that existed RF in the feed box was 41.   Therefore, the mortality probability of the rabbit was 4.08%, and the dead rabbit with RF composed 78.8% of all the dead rabbits.   It was apparent that for most of the dead rabbits, RF existed in their feed boxes before their death.   The reasons might be that the disease reduced the appetite and digestive ability of meat rabbits, and their feed intake was decreased.

During the experiment, all cases of RF appearing could be divided into three categories:

1) Case 1: Remaining feed had appeared in the feed box for several days before the rabbits died.   The number of this case was 41;

2) Case 2: RF had appeared in the feed box constantly, but the rabbits always stayed healthy.   The number of this case was 20;

3) Case 3: RF had appeared in the feed box for several days.   However, then no remaining feed appeared.   It meant that the meat rabbits were finally healthy.   The number of this case was 25.

All rabbits were divided into 4 groups according to their age at the beginning of the experiment.   Table 2 showed the RFW of the rabbits in three different cases.

Table 2   RFW of different rabbit groups in three cases

Rabbit group	Initial Age/d	Initial weight of feed Intake /g·cage^{-1}	Average weight of RF in Case 1/g	Average weight of RF in Case 2/g	Average weight of RF in Case 3/g
A	240	220	133	86	79
B	54	220	109	76	28
C	61	240	120	62	66
D	68	260	156	96	52

As shown in Table 2, the average weight of RF in Case 1 was 61.8% heavier than the average weight in Case 2, and 130.2% heavier than the average weight in Case 3.   It proved that the heavier RFW, the higher the mortality probability.   In other words, the feed intake of these unhealthy meat rabbits would decrease greatly.   In actual breeding, as the disease aggravated, the feed intake would be less and less; namely, RFW would be heavier and heavier.

According to Table 2, in Case 1 where the rabbit finally died, the average daily weight of RF in 4 rabbit groups composed 60.5%, 49.6%, 50.0%, and 60.0% of the feed quantity.   Figure 12 shows the mortality probability of the rabbit with different proportions of RFW in the feed quantity.   With the increase of the proportion of the RFW in the feed quantity, the mortality probability also increases.   When the RFW was more than 50% of feed quantity, the mortality probability of the rabbit was more than 85%; when the RFW was more than 65% of feed quantity, all the rabbits finally died in a short time.   Therefore, managers should be on the alert and manage rabbits in time if RF was observed.
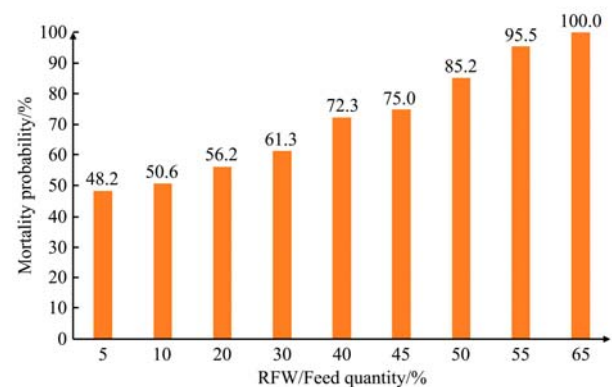

Figure 12   Relationship between RFW and mortality probability

5   Conclusions

In this study, an RFW estimation model based on the feed

instance segmentation neural network and feed weight estimation network was established. The feed instance segmentation neural network employed the framework of Mask RCNN, in which ResNet-50 was used as the backbone, PointRend algorithm was used as the network head to extract boundary. The feed weight estimation was based on BP neural network with six inputs and one output. Based on the proposed RFW estimation model, the relationship between RFW and the mortality probability of the rabbit had been explored. This technique provides a theoretical basis for non-contact rabbit health monitoring in rabbit breeding. The conclusions of this paper are shown as follows:

1) The accuracy of the proposed feed instance segmentation network was 0.997, and the total loss was 0.073. The network had a good effect on the segmentation of the mesh feed and the hollow part of the feed area. Compared with Mask RCNN, the AP value of the proposed network was 0.987, which was 4% higher; the MPA value of the proposed network was 0.985, which was 5% higher; the average running time per image of the proposed network was 0.23 s, which was 37% faster.

2) The inputs of the feed weight estimation network were extracted from the feature of the image. The Correlation Coefficient of the feed weight estimation network was 0.97, RMSE was 208.3 and RMAE was 10.51 g.

3) The RFW estimation model was used in the actual to predict RFW, and the results showed that the accuracy of the model in predicting whether RF existed was 100%, the correlation coefficient of the relationship between the predicted weight and the actual weight was 0.97.

4) The feed intake of the unhealthy meat rabbits will decrease greatly. The dead rabbit with RF composed 78.8% of all the dead rabbits, which meant that most of the unhealthy rabbits would be anorexic, and their digestibility would be reduced before they died. When the RFW was more than 50% of feed quantity, the mortality probability of the rabbit was more than 85%; When RFW was more than 65% of feed quantity, all the rabbits finally died in a short time. Therefore, it is feasible to predict the risk of the rabbit's health status by observing RFW.

Acknowledgements

[References]

[1] Cook B. Long-term monitoring of disease impact: Rabbit haemorrhagic disease as a biological control case study. The Veterinary Record, 2018; 182(20): 571–572.

[2] Cocchi M, Drigo I, Bacchin C, Bano L, Marcon B, Agnoletti F. Toxin genotyping of Clostridium perfringens strains isolated from rabbits with enteric disease. Proceedings of the 9th World Rabbit Congress, World Rabbit Science Association, Verona, Italy, 2008; pp.921–924.

[3] Agnoletti F. Update on rabbit enteric diseases: Despite improved diagnostic capacity, where does disease control and prevention stand. Proceedings 10th World Rabbit Congress, World Rabbit Science Association, Sharm El-Sheikh, Egypt, 2012; pp.3–6.

[4] Megan H, David W, Alexis G, Howard G. Biology and diseases of rabbits. Laboratory Animal Medicine (Third Edition), American College of Laboratory Animal Medicine, 2015; pp.411–461.

[5] Gu Z L. Difficulties and countermeasures for rabbits antibiotic-free breeding. Feed Industry, 2019; 40(19): 1–5. (in Chinese)

[6] Falcão-E-Cunha L, Castro-solla L, Maertens L, Marounek M, Pinheiro V, Freire J, et al. Alternatives to antibiotic growth promoters in rabbit feeding: A review. World Rabbit Science, 2007; 15(3): 127–140.

[7] Cuan K, Zhang T, Huang J, Fang C, Guan Y. Detection of avian influenza-infected chickens based on a chicken sound convolutional neural network. Computers and Electronics in Agriculture, 2020; 178: 105688. doi: 10.1016/j.compag.2020.105688.

[8] Zhang X, Kang X, Feng N, Liu G. Automatic recognition of dairy cow mastitis from thermal images by a deep learning detector. Computers and Electronics in Agriculture, 2020; 178: 105754. doi: 10.1016/j.compag. 2020.105754.

[9] Kaixuan Z, Dongjian H, Enze W. Detection of breathing rate and abnormity of dairy cattle based on video analysis. Transactions of the Chinese Society for Agricultural Machinery, 2014; 45(10): 258–263. (in Chinese)

[10] Hertem T V, Maltz E, Antler A, Romanini C, Viazzi S, Bahr C, et al. Lameness detection based on multivariate continuous sensing of milk yield, rumination, and neck activity. Journal of Dairy Science, 2013; 96(7): 4286–4298.

[11] Zotte A D. Rabbit farming for meat purposes. Animal Frontiers, 2015; 4(4): 62–67.

[12] European Commission. Commercial rabbit farming in the European Union. 2018. Available: https://op.europa.eu/en/publication-detail/-/publication/5029d977-387c-11e8-b5fe-01aa75ed71a1. Accessed on [2021-03-28].

[13] Gu Z, Qin Y, Ren K. China rabbit science. China: China Agriculture Press, 2013; pp.500–509. (in Chinese)

[14] Oglesbee B L, Lord B. Gastrointestinal diseases of rabbits. In: Ferrets, Rabbits, and Rodents, 2020; pp.174–187.

[15] Lennox A M, Kelleher S. Bacterial and parasitic diseases of rabbits. Veterinary Clinics: Exotic Animal Practice, 2009; 12(3): 519–530.

[16] Deeb B J, DiGiacomo R E. Respiratory diseases of rabbits. Veterinary Clinics: Exotic animal practice, 2000; 3: 465–480.

[17] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 2015; 28: 91–99.

[18] He K, Gkioxari G, Dollár P, Girshick Ross. Mask R-CNN. Proceedings of the IEEE international conference on computer vision, 2017; pp.2961–2969.

[19] Lin T, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection, Proceedings of the IEEE conference on computer vision and pattern recognition, 2017; pp.2117–2125.

[20] Kirillov A, Wu Y X, He K M, Girshick R. PointRend: Image segmentation as rendering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020; pp.9799–9808. doi: 10.1109/CVPR42600.2020.00982.

[21] Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines. International Conference on Machine Learning, 2010; pp.807–814. doi: 10.5555/3104322.3104425.

[22] Li J, Cheng J H, Shi J Y, Huang F. Brief introduction of back propagation (BP) neural network algorithm and its improvement. Advances in Computer Science and Information Engineering, Springer, 2012. pp.553–558. doi: 10.1007/978-3-642-30223-7_87.

[23] Jun K, Kim S J, Ji H W. Estimating pig weights from images without constraint on posture and illumination. Computers and Electronics in Agriculture, 2018; 153: 169–176.

[24] Kaili Z, Yaohong K. Neural network model and MATLAB simulation program design. China: Tsinghua University Press, 2005; pp.69–90. (in Chinese)

[25] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, 2011; 15: 315–323.

[26] Bishop C M. Pattern recognition and machine learning. Springer, 2006, 403p.

[27] Qiao Y, Truman M, Sukkarieh S. Cattle segmentation and contour extraction based on Mask R-CNN for precision livestock farming. Computers and Electronics in Agriculture, 2019; 165: 104958. doi: 10.1016/j.compag.2019.104958.

[28] Gidenne T, Combes S, Fortun-Lamothe L. Feed intake limitation strategies for the growing rabbit: effect on feeding behaviour, welfare, performance, digestive physiology and health: a review. Animal, 2012; 6: 1407–1419.