IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 23, 2019, accepted November 5, 2019, date of publication November 12, 2019,
date of current version November 22, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2953099

An Intelligent Pig Weights Estimate Method Based on Deep Learning in Sow Stall Environments

YAN CANG [1], **HENGXIANG HE** [1,2], **AND YULONG QIAO** [1]

[1] College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China
[2] Focused Loong Technology Company, Ltd, Beijing 100086, China

Corresponding author: Yan Cang (cangyan@hrbeu.edu.cn)

This work was supported in part by the National Nature Science Foundation of China under Grant 61871142, and in part by the Name was Research on Signal Processing on Graph-Based Dynamic Texture Analysis and Applications.

ABSTRACT To further the application of artificial intelligence techniques in agriculture, this study proposes an approach based on deep neural network to estimate the live weights of pigs in saw stalls. We design a neural network that uses the back of pigs in top-view depth images as the input and outputs the pig weights estimates. The proposed network, which is based on a Faster-RCNN network with an added regressive branch, integrates the pig detection and live weights regressive network into an end-to-end network. It simultaneously performs pig recognition, location and pig weights estimate. Alternating the training method optimises the proposed network. Image simulation using circles with various overlapping areas and radii is used to prove the efficacy of the proposed network. When the overlap area is greater than 30% of the total area, the proposed network is invalid. Real farm experiments were conducted for three months to construct the back of pigs in top-view depth image data set to train the proposed network. The test results not only prove the relationship between size of back area and pig weights, but also verify that the proposed neural network can accurately estimate pig weights. The study will promote the application of intelligent technique in the livestock farming, and provides some references for intelligent weighing researchers.

INDEX TERMS Digital image, deep learning, live weights estimation, object recognition.

I. INTRODUCTION

Live weight is an important parameter determining pig body condition. Changes in weights provide direct means to assess the health and growth state of the pig [1]. Furthermore, the weight of growing pigs provides a valuable parameter or indicator for keeping them at suitable level of nutrition and environment [2], [3]. Also, the commercial feeding strategies depend on the weight growing curves to increase the feed efficiency of pigs [4]. The profit from the animal is usually closely related to the balance between income and costs [5]. Accurate monitoring of weight gain performance and the use of weight data to make effective management decisions is also crucial for efficient pork production [6].

The weight of a pig is usually obtained by manual or automatic scales. The manual methods are conducted by eye and hand, based on the personal opinion of the buyer or stockman or via direct weighing of the animal [7]. Plateform scale or hanging scale methods also have been used to weigh pigs, however, the direct weighing of animals could cause injury and stress to animals and stockmen when forcing the animal onto the scale. To avoid direct contact with animals' bodies, non-contact weight methods based on computer-assisted visual images and digital images have already been developed [8], [9].

Based on the computer vision and image processing, the research on contactless weighing method system concentrates on three aspects:

1) RGB cameras are popular images collection system and widely applied to monitoring animal condition [8], [10]. Howerver, RGB cameras cannot collect the three-dimension images. Reference [11] used binocular stereo system to calculate the three-dimension size of pig body and regressive the relationship between live weight and back area in the process of growth. Moreover, depth cameras, including the Kinect [12], [13], are the substitute for binocular system to detect the three-dimension size. The pig volume are calculated from the

The associate editor coordinating the review of this manuscript and approving it for publication was Gianluigi Ciocca.

depth information and used as one of search keys in a database that lists the volume with the corresponding weights [3]. Although point clouds are used to measure body features, which are generated from depth images and more versatile than RGB or gray image [14], RGB cameras and depth cameras will coexist for a long time.

2) Finding pigs in images or videos is a very important step, no matter what kind of cameras are used. Therefore, some traditional image processing algorithms, such as edge detection, image segmentation, are used to extract the back of pigs in top-view images [9], [15], [16]. Recently, researchers tend to use deep learning algorithm to realize the pig detection [7], [17]. CNN (convolution neural network) is a popular deep neuual network and excellent in image processing, thus it has been widely applied to livestock farming [18]. Reference [19] used a short neural network, two hidden layers and a fully connected layer, to predict corresponding pig weights. There was no constraint on pig posture and image capture environment, reducing the stress of the pigs. Furthermore, reference [20] used Faster-RCNN [21], [22] network to recognize lactating sow postures. Reference [23] used Faster-RCNN network to detect the feeding behavior of group-housed pigs. It shows if Faster R-CNN network is trained by pig images, it can detect the pig accurately.

3) Since reference [24] proved that the size of the back area was closely related to pig weight, researchers focused on discussing the specific model between the pig back area and the pig weight. Reference [25] justified the argument that mixed effects model was easily adaptable to stochastic modeling. Reference [4] suggested a need for unique algorithms for specific breeds or lines of pigs. Reference [26] used a single linear regression equation to estimate the live weight of animals from the body area based on the interpretation of individual images. Reference [16] used TF model to estimate live weight with an accuracy of 97.5% on group level and 96.2% on individual level overcoming competing linear and non-linear modeling methods.

To sum up, in pig contactless weighing system, the accuracy of live weight measurement is more closely with the accuracy of the pig detection and the weight regressive model. The deep nerual network can analyse regularities in the existing data set, then accurately detect objects and predict parameters. In this study, the original structure of Faster RCNN is modificated. It proposes a deep neural network that integrates object recognition and a regression network. The input data of the proposed network is the back of pigs in top-view images captured by a depth camera, and the output is the live weight estimate of the pig. The rest of the paper is organised as: section II describes the data collection and image processing steps. Also the proposed network design is disscused, which is then followed the explanation of specific layers and the training scheme. Section III explains our experimental results and demonstrates the accuarcy on the real pig's weight estimate. Finally, section IV is the conclusion.

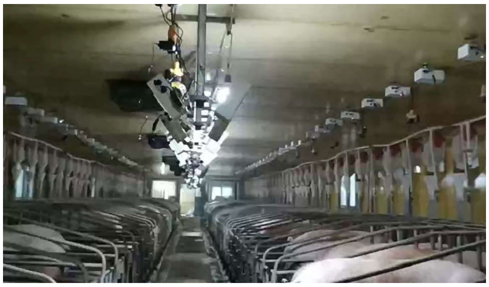

FIGURE 1. The installation of the depth camera to acquire the back of pigs in top-view images.

II. MATERIALS AND METHODS

A. ANIMALS AND TREATMENTS

The experimental pig farm is located in Jilin province, China. Twenty first farrowing sows were selected. Pigs were housed in sow stalls, shown in Fig.1. Each sow stall was installed one ground weighing scale, and pigs lived on the scale, therefore, the pig live weight was recorded all the time. Each day, the worker cleaned the ground scale to ensure the reading was the true value of the weight. Moreover, the worker calibrated the ground scale every week. Pigs were raised in a controlled temperature and humidity environment for three months. The temperature was kept on average at 22°C, with a minimum of 18.6°C and a maximum 25.4°C over the total experimental period. In the experimental period, the light in the room was natural light from 7:00 AM to 19:00 PM. The pig weight ranged from 120 kg to 200 kg.

B. DATE PREPROCESSING

The back of pigs in top-view images in the sow stall are captured by Intel D435 cameras installed in rafters of the barn. When the pig remains stationary on the ground scale, such as sleeping, the reading fluctuation of the ground scale is relatively gentle. On the contrary, when the pig is in motion, such as eating or drinking, the reading fluctuation of the ground scale is extremely large. The maximum fluctuation value can even be tens of kilograms. Aiming to the above phenomenon, only when the pig remains stationary for a long time, the automatic images acquisition system will record the reading of the ground scale, using as the ground truth. However, the system will set the camera in sleep mode to reduce redundant images. Otherwise, when the pig is in motion, the 3D camera is opened to capture the different posture images. The automatic images acquisition scheme is shown in Fig. 2, which adopts the Client/Server architecture. The server and client devices are in the same wireless LAN. The server is run in the laptop, which is connected to a hard disk. The server has a human-computer interaction interface to view the work status of each client, shown in Fig. 3. Images storage is in .PNG format.

Because each camera installation height is not exactly same, the distance independence is done as a preprocessing

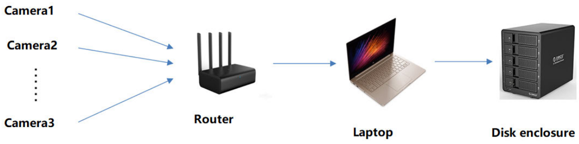

FIGURE 2. Automatic acquisition equipment deployment structure diagram.

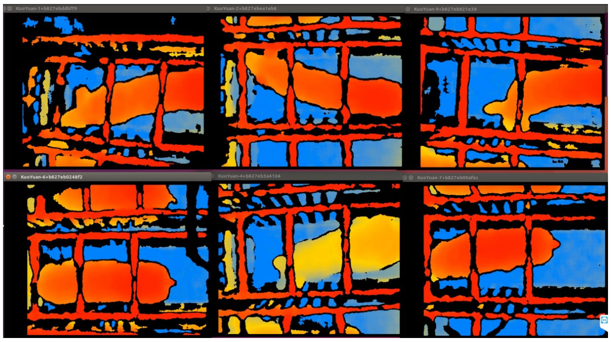

FIGURE 3. The real-time acquisition screen.

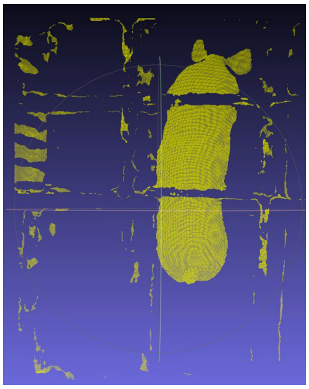

FIGURE 4. Points cloud image.

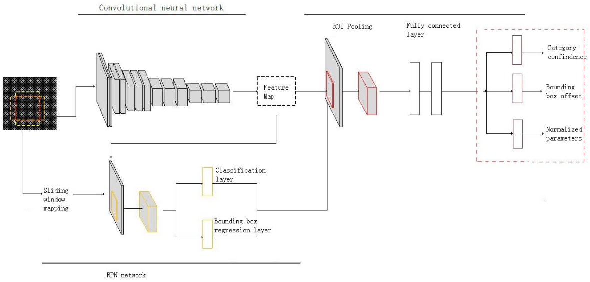

FIGURE 5. Structure of the proposed neural network.

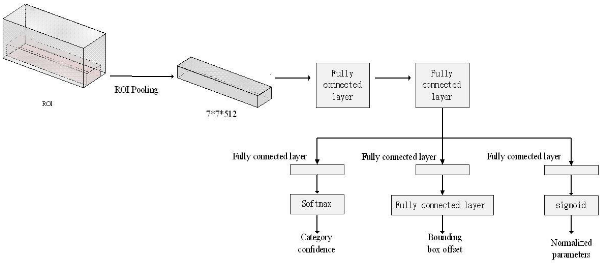

FIGURE 6. Detailed structure of the improved network.

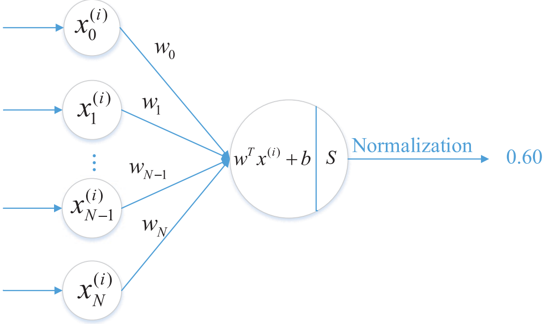

FIGURE 7. Principle of regression network implementation.

step. All depth images are transformed into points cloud space using the intrinsic parameters of cameras, followed truncated by 1.7 $m * 1.7$ $m * 1$ m. All points inside this region are mapped onto the bottom of the cuboid to generate a 2D image, which is a depth image, 8 bit, single channel, 227∗227 pixels. After distance independence processing, the points cloud is visualized in Fig. 4.

C. DEEP NEURAL NETWORK DESIGN

In this study, we introduce a regression branch into the Faster R-CNN [21], [22] to construct an end-to-end network that can recognise and locate pigs and estimate the live weight. They share the feature extraction network (Faster R-CNN), which outputs feature maps. The structure of the improved network is shown in Fig. 5.

RPN (Region Proposal Network) outputs the pig candidate bounding box of the region of interest (ROI). The feature vector of each candidate region is activated by the ReLU (Rectified Linear Unit) function and two full-connection layers. The three parallel fully-connected layers output the category confidence of the pig, the offset of the positioning box coordinates, and the normalised pig weight. Fig. 6 illustrates the details of the improved network.

ROI pooling creates a map between the proposal boxes and the vector, size 7*7*512. Two shared fully connected layers are used before the three outputs. The new regression branch is constructed by a fully connected layer and a sigmoid function. One neural cell connects the sigmoid function, which can measure all object scale parameters simultaneously (Fig. 7).

The output of the regression network is:

$$\hat{Y}^{(i)} = \frac{1}{1 + e^{-(w^T x^{(i)} + b)}} \tag{1}$$

where $\hat{Y}^{(i)}$ is the normalised estimate for the i_{th} sample and $(w^T x^{(i)} + b)$ is the feature vector of the fully-connected layer of the regression. Because the output of the sigmoid function is [0, 1], the labelled value is mapped to [0, 1] before being

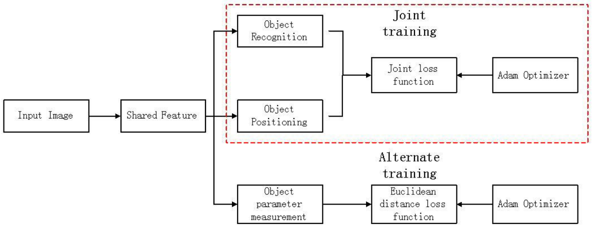

FIGURE 8. **Illustration of the improved network training scheme.**

input into the network. The normalised value is shown in (2). Equation (3) is used to estimate the parameters.

$$Y^{(i)} = \frac{y^{(i)} - y_{min}}{y_{max} - y_{min}} \tag{2}$$

$$\widetilde{y}^{(i)} = \hat{Y} * (y_{max} - y_{min}) + y_{min} \tag{3}$$

where $Y^{(i)}$ is the normalised value and $y^{(i)}$ is the true value of the i_{th} sample. y_{min} and y_{max} are the minimum and maximum of $y^{(i)}$, respectively, $\hat{Y}^{(i)}$ is the normalised value of the network, and $\widetilde{y}^{(i)}$ is the estimated value. Similar to the Faster R-CNN algorithm, when the (Intersection over Union) IOU between the label and the candidate region is less than 0.5, the ROI is considered a positive example. If and only if the ROI is positive, the loss function of the regression branch is calculated as follows:

$$L_{new} = \frac{1}{2m} \sum_{i=1}^{m} (\hat{Y}^{(i)} - Y^{(i)})^2 \tag{4}$$

where m is the total number of predicted samples, $\hat{Y}^{(i)}$ is the normalised estimated value of the i_{th} sample, and $Y^{(i)}$ is the true value of the i_{th} sample.

D. SPECIFICATION OF TRAINING SCHEMES

In this study, alternating training is used to optimise the loss functions of the regression network and the object recognition network. The specific flow-chart is shown in Fig. 8.

In the proposed neural network, although the regression network, that is the pig weight estimate network, is relatively independent, some layers of regression network are shared with the pig recognition and location network. When the regression network is trained, only coefficients of the shared layers are updated, and the weights of other networks remain unchanged. The pig recognition and location networks are used for joint training. Two Adam adaptive learning optimisers are used to optimise the loss functions. The optimiser of the Faster R-CNN is Momentum. When testing the improved network, the loss function of the regression network is not convergent and the error is larger; however, the Adam adaptive optimiser can set different learning rates according to different weighting coefficients. It is used here because it does not use too much memory and it converges within a short time. The Adam adaptive optimiser is calculated as (5) to (9).

$$m_t = \mu * m_{t-1} + (1 - \mu) * g_t \tag{5}$$

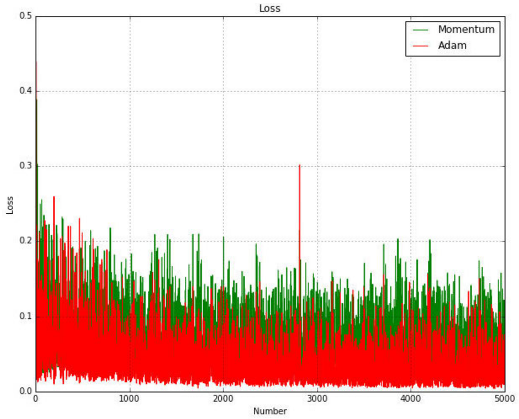

FIGURE 9. **Comparison of the loss functions of the two optimisers.**

$$n_t = \upsilon * n_{t-1} + (1 - \upsilon) * {g_t}^2 \tag{6}$$

$$\hat{m}_t = \frac{m_t}{1 - \mu^t} \tag{7}$$

$$\hat{n}_t = \frac{n_t}{1 - \upsilon^t} \tag{8}$$

$$\theta_t = \theta_{t-1} - \alpha_t * \frac{\hat{m}_t}{\sqrt{\hat{n}_t + \epsilon}} \tag{9}$$

Here, g_t is the gradient of time; m_t is the first moment estimate of a gradient and n_t is the second momentum. $\hat{m}_t$ and $\hat{n}_t$ are bias-corrected estimates of m_t and n_t. $-\frac{\hat{m}_t}{\sqrt{\hat{n}_t+\epsilon}}$ is a dynamic constraint of learning rate η, which is oscillatory in a certain range. The network parameters are optimised as follows:

$$\alpha_t = \frac{\alpha * \sqrt{1 - \beta_2^t}}{1 - \beta_1^t} \tag{10}$$

where β_1 and β_2 are hyper-parameters that control the exponential decay rate of the average gradient m_t, α is the step-size, and θ is the predicted model parameter. The default parameters are:

$$\beta_1 = 0.9 \tag{11}$$

$$\beta_2 = 0.999 \tag{12}$$

$$\epsilon = 1e - 08 \tag{13}$$

Under the same condition (same data and network parameters), a comparison of the loss function between Momentum and the Adam optimiser is shown in Fig. 9. When using the Momentum optimiser to update the network, the loss function of the regression network is shown by a green curve. When using the Adam optimiser to update the networks, the loss function is shown by a red curve. The Adam optimiser converges in a shorter time than the Momentum optimiser and the amplitude of the Adam loss function is less than that of Momentum. Therefore, the error of the Adam optimiser is smaller, making it more robust and better suited to optimise the proposed network.

TABLE 1. System environment used in the experiments.

Devices	Parameters
CPU	AMDRyzen 7 1800X Eight-Core processor*16
CPU Memory	DDR4 3000 (2600 MHz) Dual Channel 32G
GPU	GeForceGTX 1080 Ti
GPU memory	GDDR5X 11G
System Version	Ubuntu16.04
Software	TensorFlow1.3.0; CUDA8.0; cuDNN5.1; OpenCV2.4.3

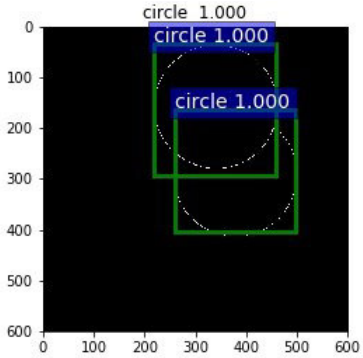

FIGURE 10. Two circles with occlusion.

E. VERIFICATION OF IMPROVED ALGORITHM PERFORMANCE

The system environment used in the experiments is shown in Table.1

F. OCCLUSION EXPERIMENTS

In this section, we use circle images to confirm the proposed network can detect the circle and estimate the radius at the same time, while we wish to verify the accuracy of radius estimation. Furthermore, in sow stalls, pig back images are always occluded by bars. Therefore, before application, we conduct a simulation to verify the relationship between the occlusion area and the accuracy of object detection. Two sample sets are generated separately, including two circles without occlusion, and two circles with occlusion. The radii are randomly selected from 50 to 150 pixels. There are 50000 images in each sample set, of which 2500 images are randomly chosen as the test set. The remaining 47,500 images construct the training set for training the object detection network. The training epoch is 20,000 times. The learning rate of the previous 1000 training values is set as 0.01, and the other learning rate is 0.001. The training batch is 256 and the training results are displayed every 20 times. In Fig.10, although the two circles are occluding each other, the occlusion area is small, and the object detection network recognises and locates them clearly. Conversely, in Fig.11, there is a larger overlap area between the two circles, and the object detection network recognises them as one complete circle.

The test results of 2500 samples are summarised in Table 2. The AP is the mean Average Precision. Statistically, occlusion between two circles increases the miss rate to 15.04%. To determine how the occlusion area affects the recognition accuracy, we generate 25 datasets with an overlap rate from 10% to 40%. The difference in radii varies from 0 to 50 pixels. There are 8,000 samples in the training set and 2000 samples in the test set. Each dataset is trained 20,000 times, resulting in 25 models. Tables 3, 4, and 5 show the object detection results for a variable overlap rate. The average testing time of one image is 0.143s. The results indicate that, regardless of the radius difference, if the object overlap rate is greater than 30%, the missed detection rate increases rapidly, with an average miss rate of 47.73% . The accuracy of the object detection network also decreases rapidly when the object overlap rate is over 30%. Therefore, in the dataset used in the regression experiment, the overlap ratios are all less than 30%.

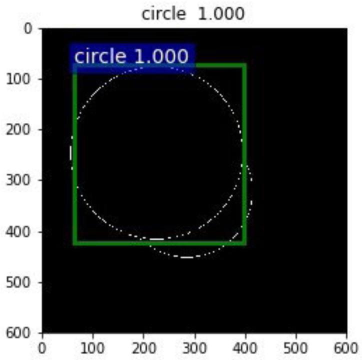

FIGURE 11. Two circles with larger occlusion area.

TABLE 2. Object recognition results of 2500 sample.

	Without occlusion	With occlusion
AP(%)	1.000	0.909
Missed rate(%)	0.00	0.75

TABLE 3. Object detection results for an equal circle radius.

Overlap rate(%)	10	20	30	40
AP(%)	1.000	0.0901	0.7273	0.5455
Missed rate(%)	0.00	1.7	51.1	99.6

TABLE 4. Object detection results for a radius difference of 10.

Overlap rate(%)	10	20	30	40
AP(%)	1.000	0.0901	0.7273	0.5455
Missed rate(%)	0.00	0.4	47.9	99.7

TABLE 5. Object detection results for a radius difference of 40.

Overlap rate(%)	10	20	30	40
AP(%)	1.000	0.0901	0.7273	0.5455
Missed rate(%)	0.00	0.1	44.2	71.3

G. RADIUS MMEASUREMENT OF A SINGLE CIRCLE

In this experiment, we generate 50,000 single circles. The data are divided into 47500 training samples and 2500 test samples. The single circle has a radius of [50, 150]. The number of training instances is 20000, the network training

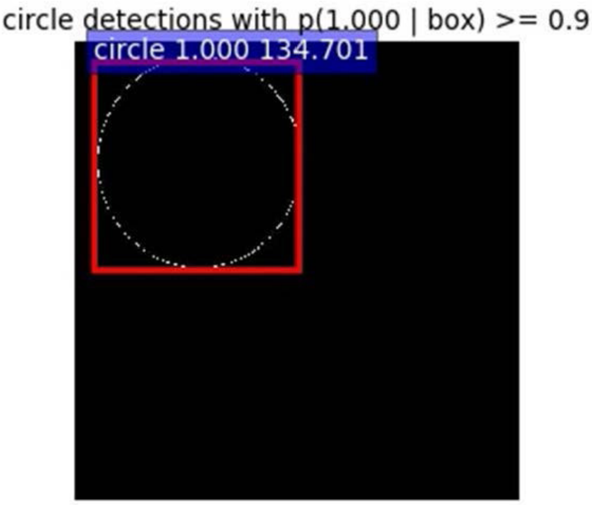

FIGURE 12. Measurement results for a single circle.

TABLE 6. Measurement results of single circle radius.

Image index	Ground truth(pixel)	Measurement (pixel)	Absolute error(pixel)	Relative error(%)
000046	50.000	52.176	2.176	4.352%
007130	51.000	52.514	1.514	2.969%
001399	64.000	66.273	2.273	3.552%
016888	68.000	69.218	1.218	1.791%
015785	70.000	70.941	0.941	1.344%
017547	75.000	76.970	1.970	2.627%
017642	81.000	80.758	0.242	0.299%
002714	86.000	87.107	1.107	1.287%
001438	91.000	90.064	0.936	1.029%
019775	99.000	97.939	1.061	1.072%
016471	106.000	107.251	1.251	1.180%
004369	110.000	107.670	2.330	2.118%
002795	114.000	111.607	2.393	2.099%
010537	119.000	115.132	3.868	3.250%
014770	121.000	122.615	1.615	1.335%
004782	129.000	128.383	0.617	0.478%
011773	131.000	130.453	0.547	0.418%
017931	137.000	136.654	0.346	0.253%

batch is 256, and the initial learning rate is 0.00001, which decays by 10 times per 10,000 training instances. The training uses the GPU mode. The radius estimate results of single circle images are shown in Fig. 12. The outputs are the object (circle), the confidence (1.000), and the radius estimate (134.701). Some of the radius estimation results are shown in Table 6, which lists the radius estimate of the circle, the ground truth of the radius, the absolute error, and the relative error. The ten statistical results show the average relative error of the network measurement is 1.55%, and the average variance of the measurement average relative error is 1.24. This experiment proves that the proposed algorithm can recognise the object and simultaneously predict the scale parameter of the object.

H. RADIUS MEASUREMENT OF TWO CIRCLES WITH OCCLUSION

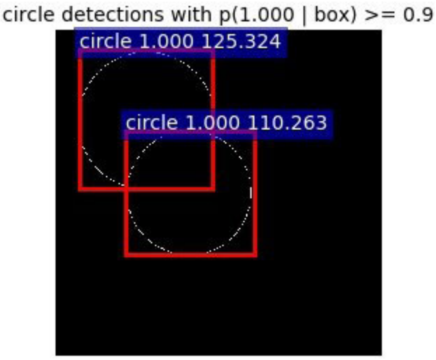

FIGURE 13. Measurement results for two circles.

TABLE 7. Measurement results of the radius of two circles with occlusion.

Image index	Ground truth(kg)	Measurement (kg)	Absolute error(kg)	Relative error(%)
000011	97.000	96.177	0.823	0.848
000011	70.000	73.717	3.717	5.310
000031	142.000	139.927	2.073	1.460
000031	100.000	98.903	1.097	1.097
000037	116.000	112.735	3.265	2.815
000037	144.000	142.400	1.600	1.111
000058	107.000	105.661	1.339	1.251
000058	72.000	72.679	0.679	0.943
000081	131.000	129.253	1.747	1.334
000081	100.000	100.397	0.397	0.397
000226	100.000	98.900	1.100	1.100
000226	56.000	57.099	1.099	1.962
000347	58.000	59.950	1.950	3.362
000347	71.000	71.259	0.259	0.365
000672	111.000	109.831	1.169	1.053
000672	127.000	125.672	1.328	1.046
000850	57.000	58.487	1.487	2.609
000850	83.000	83.882	0.882	1.063
000960	130.000	130.290	0.290	0.223
000960	144.000	141.727	2.273	1.578

A total of 50,000 images are randomly generated, which an overlap rate of two circles is less than 30% and a radius difference is less than 50 pixels. 47500 training samples and 2500 test samples are used. Experiment parameters and environment settings are identical to those of the single circle experiment. The radius measurement results of two circle images are shown in Fig. 13. The outputs are the object (circles), the confidence (1.000), and the radius estimate (125.324 and 110.263). To clearly illustrate the effectiveness of the improved algorithm, some of the radius estimation results are shown in Table 7. 2500 test images, the average relative error is 1.64% and the average variance of the predicted average relative error is 1.45. The improved algorithm extends the application scope of the original Faster R-CNN algorithm and the simulation results prove the efficiency of the improved network. This experiment proves that with occulsion the proposed algorithm still can recognise the object and simultaneously predict the scale parameter of the object.

III. RESULTS OF PIG LIVE WEIGHT EXPERIMENTS

A. DATASET

Based on the simulation results of the circle images, the proposed neural network was applied to the pig contactless

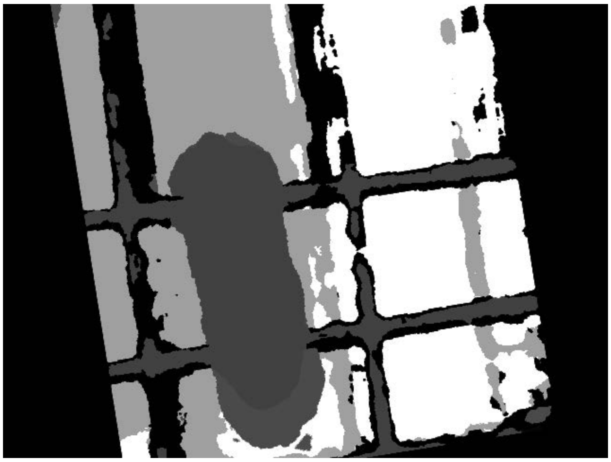

FIGURE 14. Perfect back image.

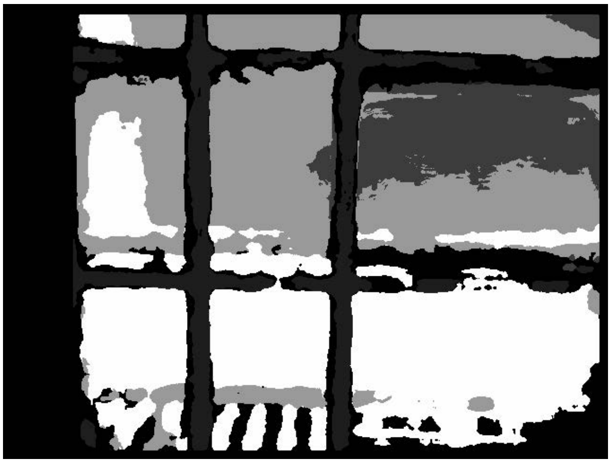

FIGURE 15. Bad back image.

weighing. The experimental dataset comprises depth images, which were the back of pigs in top-view images captured by a depth camera. The sample dataset included two groups of images: perfect back images and poor back images, which were shown in Fig. 14,15.

The grey scale value represented the distance between the back of the pig and the camera. White colours indicated that the object was far from the camera, whereas black colours indicated that the object was close to the camera. The grids were a projection of the bars of the sow stall. As could be seen in Fig. 14, the outline of the pig's back was distinct and the occlusion area was less than 30%. In Fig. 15, because of movement or other reasons, only the projection of the bars was observed and the back of the pig could not be determined; therefore, the image was poor. All images in the training set were labelled as 'pig' or 'pig_bad' by hand. The pig weight was labelled using the ground scale reading. Labelling software labelled the images into .XML files, which included the image index, image size, object category or background, position coordinates, and the ground truth. In this experiment, we wrote the ground scale reading into the image file name so that the ground truth of the pig weight was read with the file name. The rule of naming images was: index_weight_camera.png; for example: 000001_164.68_0.png. In the experiment, we labelled 12718 images, including 9989 training images and 2729 test images. The input image size was set to 600*800 pixels. To avoid under fitting, the training dataset was enlarged by mirrors and rotation. The axis of symmetry was the image centre line. After mirror and rotation processing, the total number of images was 19978. When rotating the image, the ROI experienced the same rotation. The object proposals also exhibited a one-to-one match with the label. In the training set, the true pig weight range was from 159.27 kg to 167.27 kg.

B. INITIAL NETWORK AND TRAINING PARAMETERS

Before training, all pig weight values were normalised and mapped in the range of [0, 1]. The object recognition network outputted three categories: the pig, the pig_bad, and the background. The neuronal cell number of the object location network was set to 3*4. The number of neuronal cells of the pig weight estimation network was set to 1 and the output of the network was the normalised pig weight estimate. Because the proposed network only used the previous 13 levels of the (Visual Geometry Group) VGG network, the transfer learning only involved these 13 levels. Weight coefficients of the basic feature extraction network did not update, so the previous four levels of weight parameters were unaltered (conv1_1;conv1_2;conv2_1 and conv2_2) and the weight coefficients of the other levels were updated in terms of the input images. The RPN network and Faster RCNN convolution parameters initially used the Gaussian function with a zero mean value and a standard variance of 0.01. The training batch was 256, the training iteration was 20000 time, the learning rate was 0.0001, and the training was in GPU mode.

C. EXPERIMENTAL RESULTS

There were 2729 images in the test dataset. To prove that the model generalisation performance was satisfied, the test data and training data were completely different. Table 8 showed 20 pig weight estimates distributed from 159.27 kg to 167.27 kg. The average execution time of one image was 0.174s.

To more clearly analyse the differences between the true values and the estimates, we randomly chose 1000 images and presented the errors in Fig. 16. The green and red lines represented the absolute and relative errors, respectively. The absolute error was typically less than 0.8 kg and the relative error was less than 0.5%. The relative error was close to 0.5%. The average absolute error of 2729 test images was 0.644 kg and the average relative error was 0.374%.

Fig.17–19 showed four examples of the sow stall application. The model outputted three parameters: pig (category), probability, and weight. The AP was 0.909. In example 1, although the pig's back was curved, the proposed network recognised the pig and provided a weight

TABLE 8. Pig weight estimates.

Image index	Ground truth(kg)	Measurement (kg)	Absolute error (kg)	Relative error(%)
1	159.27	159.397	0.127	0.080
2	159.44	159.571	0.131	0.082
3	159.61	159.796	0.186	0.117
4	159.94	159.990	0.050	0.031
5	160.11	159.983	0.127	0.079
6	160.62	160.322	0.298	0.186
7	160.79	160.436	0.354	0.220
8	161.31	161.892	0.582	0.361
9	161.80	161.081	0.719	0.444
10	161.97	161.737	0.233	0.144
11	162.13	161.742	0.388	0.239
12	162.27	162.057	0.213	0.131
13	162.30	161.523	0.777	0.479
14	162.47	162.326	0.144	0.089
15	162.64	161.463	1.177	0.724
16	162.72	162.024	0.696	0.428
17	162.81	162.253	0.557	0.342
18	162.98	162.927	0.053	0.033
19	163.12	162.364	0.756	0.463
20	163.15	162.815	0.335	0.205

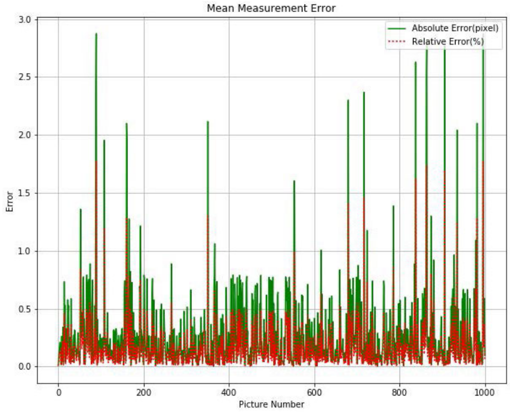

FIGURE 16. Absolute error and relative error.

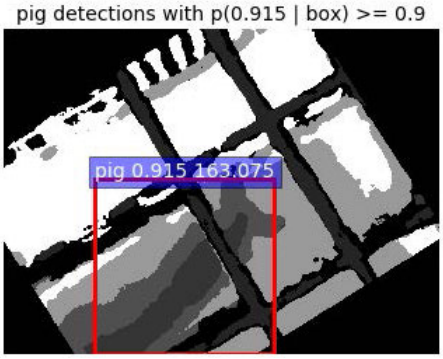

FIGURE 17. Example 1.

estimate of 163.075 kg, and the true weight was 163.80 kg. In example 2, the pig head was not in the images, only the back, but the proposed network recognised the pigs successfully and estimated the pig weight as 139.678 kg and the ture weight was 139.00 kg. Example 3 showed a perfect pig back image, and the proposed network estimated the weight as 160.820 kg and the true weight was 160.5 kg. Therefore, the proposed network had learned the potential relationship between pig back images and pig weight. The proposed network precisely recognised and located pigs and accurately predicted pig weight when the overlap area in the image was less than 30%.

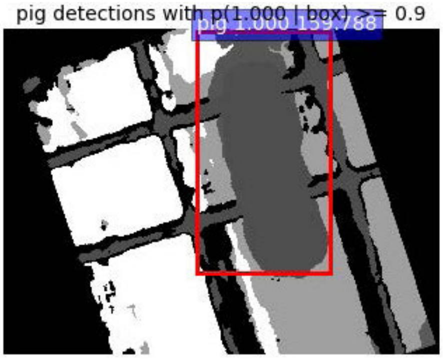

FIGURE 18. Example 2.

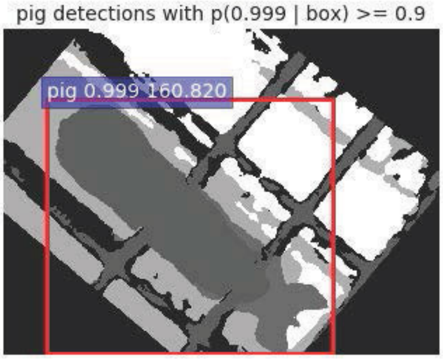

FIGURE 19. Example 3.

IV. CONCLUSION

In this study, we improve the network structure of the Faster R-CNN object detection algorithm by introducing a regression neural network and apply it to pig weight estimate using depth images. The new network simultaneously achieves pig recognition, pig location and pig weight estimate. The alternating training scheme achieves rapid convergence of the network and the Adam optimiser replaces the Momentum optimiser. Circle image experiments show that the average relative error of a single circle radius estimate is 1.55%. When there is less than a 30% overlap between two circles, the proposed network can predict the two radii with an average relative error of 1.64%. Therefore, the improved network exhibits good regressive performance. The improved neural network is then used to evaluate pig weight by inputting the back of pigs in top-view images. The network is trained in an alternating training scheme. The average absolute error of the pig weight estimates is 0.644 kg and the relative error is 0.374%. The proposed deep neural network can therefore be used to estimate pig weight using the direct relationship between pig weight and the back of pigs in top-view depth images. The added regressive branch used in this study shares a general feature extraction network with the Faster RCNN algorithm. In conclusion, the proposed deep learning algorithm can be used to develop a non-contact pig weighing system.

In future, aimming to group housed pigs' live weight estimation, the instance segmentation network will be introduced. The image collector will collect top-view depth images, involving multiple pigs. The segmentation network will divid them into multiple independent pigs' back images. By transfer learning of the sow live weigth estimation network, the fattening pigs' live weight estimation can get the satified accuracy. However, the different posture of pigs will affect the weight accuracy, which will be disscused furtherly.

REFERENCES

[1] S. L. Douglas, O. Szyszka, K. Stoddart, S. A. Edwards, and I. Kyriazakis, "Animal and management factors influencing grower and finisher pig performance and efficiency in European systems: A meta-analysis," *Animal*, vol. 9, no. 7, pp. 1210–1220, 2015, doi: 10.1017/s1751731115000269.

[2] S. G. Matthews, A. L. Miller, J. Clapp, T. Plötz, and I. Kyriazakis, "Early detection of health and welfare compromises through automated detection of behavioural changes in pigs," *Vet. J.*, vol. 217, pp. 43–51, Jan. 2016, doi: 10.1016/j.tvjl.2016.09.005.

[3] K. Wang, H. Guo, Q. Ma, W. Su, L. Chen, and D. Zhu, "A portable and automatic Xtion-based measurement system for pig body size," *Comput. Electron. Agricult.*, vol. 148, pp. 291–298, May 2018, doi: 10.1016/j.compag.2018.03.018.

[4] A. V. Fisher, D. M. Green, C. T. Whittemore, J. D. Wood, and C. P. Schofield, "Growth of carcass components and its relation with conformation in pigs of three types," *Meat Sci.*, vol. 65, no. 1, pp. 639–650, 2003, doi: 10.1016/S0309-1740(02)00266-8.

[5] D. M. Green, S. Brotherstone, C. P. Schofield, and C. T. Whittemore, "Food intake and live growth performance of pigs measured automatically and continuously from 25 to 115 kg live weight," *J. Sci. Food Agricult.*, vol. 83, no. 11, pp. 1150–1155, 2010, doi: 10.1002/jsfa.1519.

[6] D. J. Parsons, D. M. Green, C. P. Schofield, and C. T. Whittemore, "Real-time control of pig growth through an integrated management system," *Biosyst. Eng.*, vol. 96, no. 2, pp. 257–266, 2007, doi: 10.1016/j.biosystemseng.2006.10.013.

[7] A. Wongsriworaphon, B. Arnonkijpanich, and S. Pathumnakul, "An approach based on digital image analysis to estimate the live weights of pigs in farm environments," *Comput. Electron. Agricult.*, vol. 115, pp. 26–33, Jul. 2015, doi: 10.1016/j.compag.2015.05.004.

[8] P. Ahrendt, T. Gregersen, and H. Karstoft, "Development of a real-time computer vision system for tracking loose-housed pigs," *Comput. Electron. Agricult.*, vol. 76, no. 2, pp. 169–174, 2011, doi: 10.1016/j.compag.2011.01.011.

[9] W. Huang, W. Zhu, C. Ma, Y. Guo, and C. Chen, "Identification of group-housed pigs based on Gabor and Local Binary Pattern features," *Biosyst. Eng.*, vol. 166, pp. 90–100, Feb. 2018, doi: 10.1016/j.biosystemseng.2017.11.007.

[10] W. X. Zhu, Y.-Z. Guo, P.-P. Jiao, C.-H. Ma, and C. Chen, "Recognition and drinking behaviour analysis of individual pigs based on machine vision," *Livestock Sci.*, vol. 205, pp. 129–136, Nov. 2017, doi: 10.1016/j.livsci.2017.09.003.

[11] C. Shi, G. Teng, and Z. Li, "An approach of pig weight estimation using binocular stereo system based on LabVIEW," *Comput. Electron. Agricult.*, vol. 129, pp. 37–43, Nov. 2016, doi: 10.1016/j.compag.2016.08.012.

[12] C. Chen, W. X. Zhu, Y. Z. Guo, C. H. Ma, W. J. Huang, and C. Z. Ruan, "A kinetic energy model based on machine vision for recognition of aggressive behaviours among group-housed pigs," *Livestock Sci.*, vol. 218, pp. 70–78, Dec. 2018, doi: /10.1016/j.livsci.2018.10.013.

[13] J. Kongsro, "Estimation of pig weight using a Microsoft Kinect prototype imaging system," *Comput. Electron. Agricult.*, vol. 109, pp. 32–35, Nov. 2014, doi: 10.1016/j.compag.2014.08.008.

[14] K. Jun, S. J. Kim, and H. W. Ji, "Estimating pig weights from images without constraint on posture and illumination," *Comput. Electron. Agricult.*, vol. 153, pp. 169–176, Oct. 2018, doi: 10.1016/j.compag.2018.08.006.

[15] M. A. Kashiha, C. Bahr, S. Ott, C. P. H. Moons, T. A. Niewold, F. O. Ödberg, and D. Berckmans, "Automatic weight estimation of individual pigs using image analysis," *Comput. Electron. Agricult.*, vol. 107, pp. 38–44, Sep. 2014, doi: 10.1016/j.compag.2014.06.003.

[16] M. A. Kashiha, C. Bahr, S. Ott, C. P. H. Moons, T. A. Niewold, F. Tuyttens, and D. Berckmans, "Automatic monitoring of pig locomotion using image analysis," *Livestock Sci.*, vol. 159, no. 1, pp. 141–148, Jan. 2014, doi: 10.1016/j.livsci.2013.11.007.

[17] K. Juna, S. J. Kimb, and H. W. Jic. "Estimating pig weights from images without constraint on posture and illumination," *Comput. Electron. Agricult.*, vol. 153, pp. 169–176, 2018.

[18] R. Alvarez, J. Arroqui, P. Mangudo, J. Toloza, D, Jatip, J. M. Rodríguez, A. Teyseyre, C. Sanz, A. Zunino, C. Machado, and C. Mateos, "Body condition estimation on cows from depth images using Convolutional Neural Networks," *Comput. Electron. Agricult.*, vol. 155, no. 2, pp. 12–22, 2018, doi: 10.1016/j.compag.2018.09.039.

[19] K. Jun, S. J. Kim, and H. W. Ji, "Estimating pig weights from images without constraint on posture and illumination," *Comput. Electron. Agriculture*, vol. 153, pp. 169–176, 2018, doi: 10.1016/j.compag.2018.08.006.

[20] C. Zheng, X. M. Zhu, X. Yang, L. Wang, S. Tu, and Y. Xue, "Automatic recognition of lactating sow postures from depth images by deep learning detector," *Comput. Electron. Agricult.*, vol. 147, pp. 51–63, Apr. 2018, doi: 10.1016/j.compag.2018.01.023.

[21] R. Girshick, "Fast R-CNN," *Comput. Sci.*, vol. 153, pp. 169–176, Apr. 2015. [Online]. Available: https://arxiv.org/abs/1504.08083

[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[23] Q. Yang, D. Xiao, and S. Lin, "Feeding behavior recognition for group-housed pigs with the faster R-CNN," *Comput. Electron. Agricult.*, vol. 155, pp. 453–460, Dec. 2018, doi: 10.1016/j.compag.2018.11.002.

[24] C. P. Schofield, "Evaluation of image analysis as a means of estimating the weight of pigs," *J. Agricult. Eng. Res.*, vol. 47, pp. 287–296, Sep./Dec. 1990, doi: 10.1016/0021-8634(90)80048-Y.

[25] B. A. Craig and A. P. Schinckel, "Nonlinear mixed effects model for swine growth," *Prof. Animal Scientist*, vol. 17, no. 4, pp. 256–260, 2001, doi: 10.15232/S1080-7446(15)31637-5.

[26] T. H. Liu, "Prediction of pig weight based on radical basis function neural network," *Trans. Chin. Soc. Agricult. Machinery*, vol. 44, no. 8, pp. 245–249, 2013, doi: 10.6041/j.issn.1000-1298.2013.08.042.

YAN CANG was born in Harbin, China, in 1978. She received the B.S. degree in material engineering from the University of Jiamusi, Heilongjiang, China, in 2000, and the M.S. degree in power engineering and the Ph.D. degree in information and communication engineering from Harbin Engineering University, Harbin, China, in 2003 and 2008, respectively.

She was a Lecture with Harbin Engineering University. She holds one patent. Her research interests include artificial intelligent algorithm and image signal processing.

HENGXIANG HE was born in Zhoushan, Zhejiang, China, in 1991. He received the B.S. and M.S. degrees in information and communication engineering from Harbin Engineering University, Heilongjiang, China, in 2015, where he is currently pursuing the Ph.D. degree in information and communication engineering. He did part-time job with Focused Loong Technology Company, Ltd., in 2017.

His research interests include the distributed computation, artificial intelligent algorithm, and image signal processing.

YULONG QIAO was born in Taiyuan, Shanxi, China, in 1977. He received the B.S. and M.S. degrees in applied mathematics and the Ph.D. degree from the Harbin Institute of Technology, Harbin, Heilongjiang, China.

He is currently a Professor with Harbin Engineering University. His current research interests include texture analysis, image and video processing, signal processing on graph, and pattern recognition.