Chapter 51
Pig Pose Recognition Method Based on Openpose

Guangchang Li, Qing Jv, Feifei Liu, and Zhibo Yao

Abstract Early detection and treatment of disease in pigs can increase pork production. The changes of pig posture information have high diagnostic value, but these subtle changes are not easy to be quantified and need to be observed and analyzed for a long time, which is difficult to achieve in the process of large-scale breeding. In this paper, a pig pose recognition method based on Openpose is proposed for pig pose recognition. Firstly, the collected images were input into the OpenPose pose estimation model to obtain the key point location information and key point connection information of pigs. The two groups of information were converted into joint angle features and joint spacing features, which were input into the KNN algorithm to classify pig poses. The experimental results show that the accuracy of the model for pig attitude recognition is more than 93%, and the FPS is between 10 and 12, which is a method with both accuracy and speed.

51.1 Introduction

In recent years, affected by African Swine fever and other infectious diseases of pigs, the price of pork skyrocketed, seriously affecting people's daily life. Therefore, early detection of damage to the health of pigs and timely treatment can increase pork production so as to improve people's livelihood.

G. Li · Q. Jv · F. Liu (✉) · Z. Yao
School of Mechanical and Electrical Engineering, Jiangxi University of Science and Technology, Ganzhou, China
e-mail: gzlff@126.com

G. Li
e-mail: 2359795949@QQ.COM

Q. Jv
e-mail: 940680576@QQ.COM

Z. Yao
e-mail: 1406394534@QQ.COM

As a kind of easily accessible biological information, pig posture information is an important indicator to measure the health status of pigs. However, the changes of posture information is not easy to be quantified and requires long-term observation by staff, which is difficult to achieve in the process of large-scale breeding. Therefore, the objective monitoring of the change of attitude information through sensors has attracted wide attention [1]. Based on the analysis of pixel difference between continuous images, a method of automatic detection of aggressive behavior between pigs by image analysis was proposed [2]. The Kinect depth camera was used to extract the behavior information of pigs, and the support vector machine was used to detect attack behavior. This method was cost-effective and effective in terms of accuracy [3].

In this paper, the OpenPose pose estimation algorithm [4] is used to identify the key points of pigs, the position and connection information of the key points are obtained, then the two groups of information are converted into the bone joint angle and joint spacing of pigs, and then the KNN [5] algorithm was used classification and recognition of pig posture.

51.2 Materials and Methods

51.2.1 Data Preparation

The experimental data came from a pig breeding base in Ganzhou city, Jiangxi Province. Cameras were used to shoot the video at different angles and with different degrees of occlusion. And 4000 images were selected as the training and testing sample data. The image size was adjusted to 360 pixels × 520 pixels to improve the training speed of the model.

Labelme image annotation tool was used to annotate the key points and pose categories of pigs. Finally, Json files with information were used. Key points and pose category comments are shown in Fig. 51.1.

See Fig. 51.2, the selected image is rotated, mirrored, and scaled to enhance and expand the sample image.

51.2.2 Pig Pose Estimation

In actual breeding, multiple pigs are raised in a pig pen, so the OpenPose pose estimation algorithm is adopted in this paper to carry out real-time pose estimation for multiple pigs. The OpenPose algorithm adopts bottom-up approach and has achieved good results in real-time pose estimation [4].
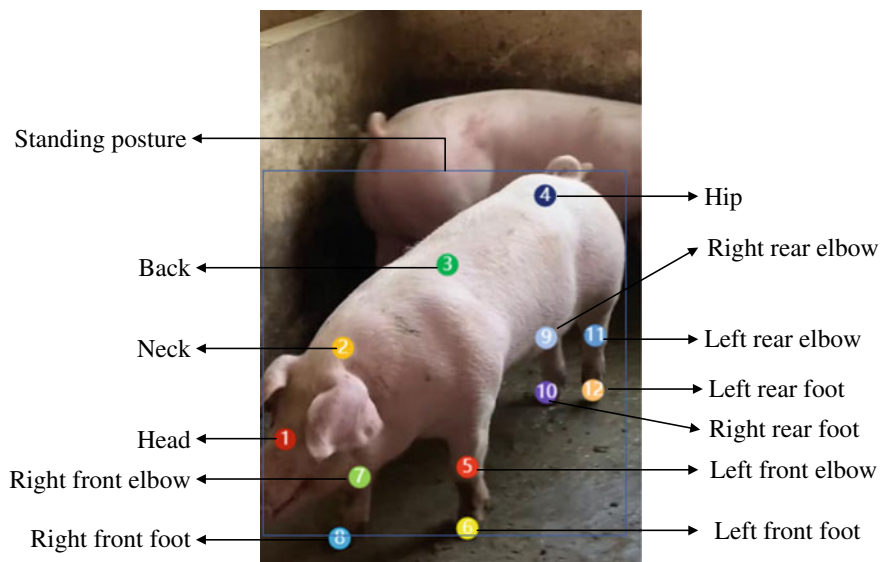
Standing posture
Hip
Right rear elbow
Back
Left rear elbow
Neck
Left rear foot
Right rear foot
Head
Right front elbow
Left front elbow
Right front foot
Left front foot

Fig. 51.1 Picture annotation schematic diagram

Original

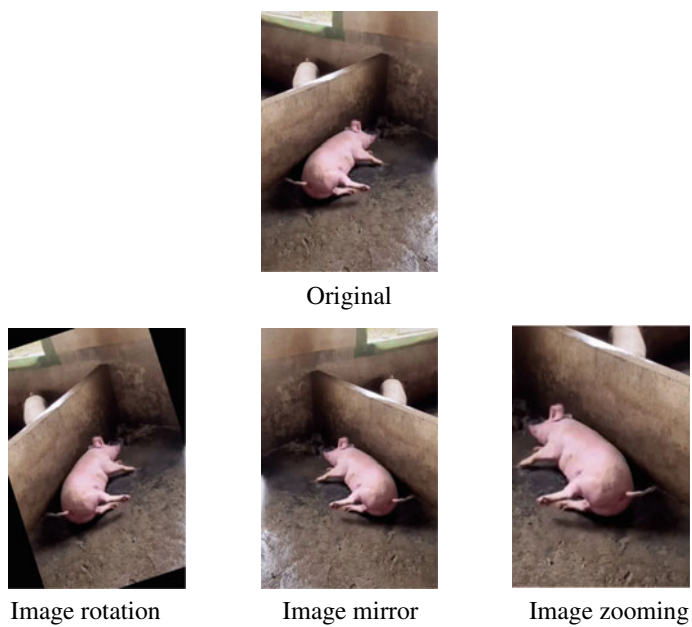Image rotation　　　　Image mirror　　　　Image zooming
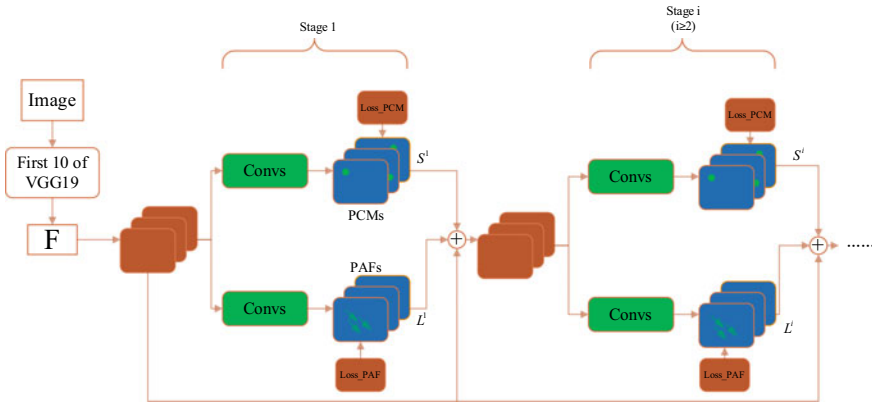
Fig. 51.2 Data to enhance

Fig. 51.3 The OpenPose Network architecture

OpenPose Network Architecture

The OpenPose network generates Part confidence maps (PCMs) and Part affinity fields (PAFs) based on the input images. The former is used to predict the location of key points, and the latter is used to group key points and finally obtain the attitude skeleton. The network architecture of OpenPose is shown in Fig. 51.3.

The first 10 layers of VGG19 [6] network were used for feature extraction, and the feature images were input to the subsequent Stage module for processing. The Stage module contains two branches, S and L. The S branch is used to generate the confidence diagram of the key point, and the L branch is used to generate the affinity domain of the key point. L2 Loss is used to calculate the Loss of each branch, and the Loss function of the branch is shown in the formula.

$$
\begin{aligned}
f_{\mathbf{S}}^i &= \sum_{j=1}^{J} \sum_{\mathbf{p}} \mathbf{W}(\mathbf{p}) \cdot \left\| \mathbf{S}_j^i(\mathbf{p}) - \mathbf{S}_j^*(\mathbf{p}) \right\|_2^2, \\
f_{\mathbf{L}}^i &= \sum_{c=1}^{C} \sum_{\mathbf{p}} \mathbf{W}(\mathbf{p}) \cdot \left\| \mathbf{L}_c^i(\mathbf{p}) - \mathbf{L}_c^*(\mathbf{p}) \right\|_2^2.
\end{aligned}
\tag{51.1}
$$

Among them, $W(p)$ represents the weight of the key point. If the position is not marked, 0; otherwise, 1; $\mathbf{L}_c^*(\mathbf{p})$ and $\mathbf{S}_j^*(p)$ represents the predicted value; $\mathbf{L}_c^i(\mathbf{p})$ and $\mathbf{S}_j^i(p)$ represents the true value. The total loss is the sum of the losses of each branch, as shown in the formula (51.2),

$$
f = \sum_{t=1}^{T} \left(f_{\mathbf{S}}^i + f_{\mathbf{L}}^i \right)
\tag{51.2}
$$

Starting from the second Stage module, the input of the module consists of three parts, the feature diagram F, the output of S^{i-1}, and the output of L^{i-1}, as shown in the formula (51.3).

$$
\begin{aligned}
\mathbf{S}^i &= \rho^i\left(\mathbf{F}, \mathbf{S}^{i-1}, \mathbf{L}^{i-1}\right), \forall i \geq 2, \\
\mathbf{L}^i &= \phi^i\left(\mathbf{F}, \mathbf{S}^{i-1}, \mathbf{L}^{i-1}\right), \forall i \geq 2.
\end{aligned}
\tag{51.3}
$$

OpenPose components

Part confidence maps are used to represent the position of the key point. The Gaussian function is used to create the confidence graph of the key point. The value of each pixel in the confidence graph represents the confidence degree of the position, which is calculated by the formula (51.4).

$$
g(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{\left((x-x')^2+(y-y')^2\right)}{2\sigma^2}}.
\tag{51.4}
$$

In this paper, the PCMs output contains 12 porcine key point location information and one containing background information. The background information serves as input to the next Stage, where better semantic information can be obtained. When σ set to 0.008, the output looks like Fig. 51.4.

Channel 0: PCM of Head   Channel 1: PCM of Neck
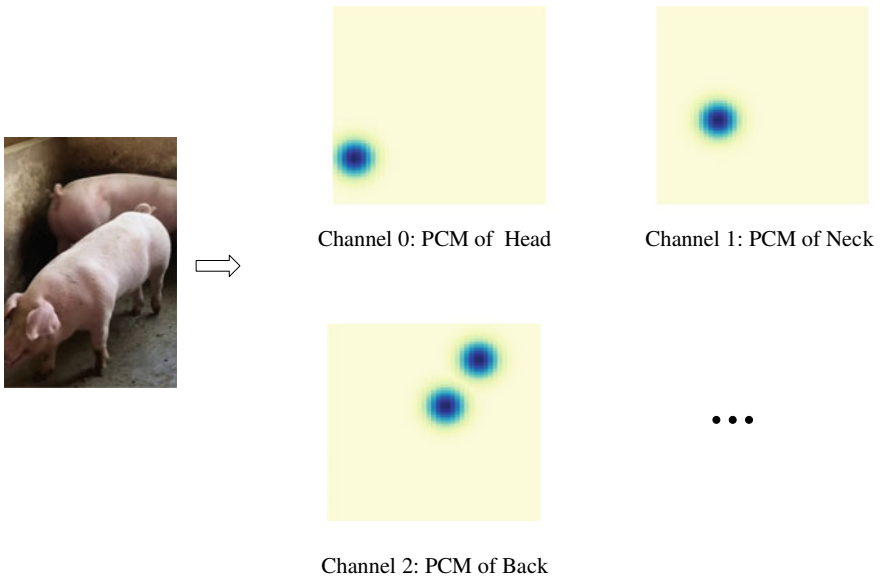
Channel 2: PCM of Back

Fig. 51.4 PCMs output. There are two regions with high confidence in the channel 2, so it can be judged that there are two pigs in the input image
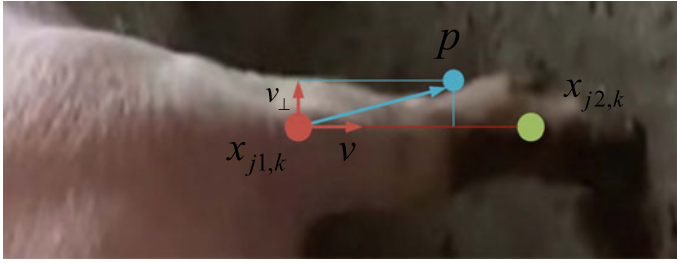
Fig. 51.5 Left front leg of pig

Part affinity fields are used to judge the affinity between different key points. Each pixel in the PAF region holds a two-dimensional vector, which is used to represent the directional information of the bone. Take the left front leg as an example, as shown in Fig. 51.5. $x_{j1,k}$ and $x_{j2,k}$ respectively represent the position coordinates of the left front elbow and the left front hoof.

For position p, its vector is expressed by the formula (51.5).

$$L_{c,k}^{*}(\mathbf{p}) = \begin{cases} \mathbf{v}, \textbf{ if } p \textbf{ on limb } c, k \\ \mathbf{0}, \textbf{ otherwise} \end{cases}. \tag{51.5}$$

c is the index of the limb, and $\mathbf{v} = \frac{(x_{j2,k}-x_{j1,k})}{\|x_{j2,k}-x_{j1,k}\|_{2}}$ represents the pointing unit vector. For different positions use formula (51.6) to determine whether it is on the limb.

$$\begin{aligned} 0 &\leq \mathbf{v} \cdot (p - x_{j1,k}) \leq l_{c,k} \\ &\textbf{and } \big|\mathbf{v} \times (p - x_{j1,k})\big| \sigma_{l}, \end{aligned} \tag{51.6}$$

where $l_{c,k} = \|x_{j2,k} - x_{j1,k}\|_{2}$ indicates the length of the limb, and σ_{l} indicates the width of the limb. For any two key points $\mathbf{d}_{j_1}$ and $\mathbf{d}_{j_2}$, the affinity between the key points can be calculated using formula (51.7),

$$E = \int_{u=0}^{u=1} L_{c}(\mathbf{p}(u)) \cdot \frac{\mathbf{d}_{j_2} - \mathbf{d}_{j_1}}{\|\mathbf{d}_{j_2} - \mathbf{d}_{j_1}\|_{2}} du. \tag{51.7}$$

$\mathbf{p}(u)$ is obtained by position and interpolation, and the calculation is shown in the formula (51.8)

$$\mathbf{p}(u) = (1 - u)\mathbf{d}_{j_1} + u\mathbf{d}_{j_2}. \tag{51.8}$$

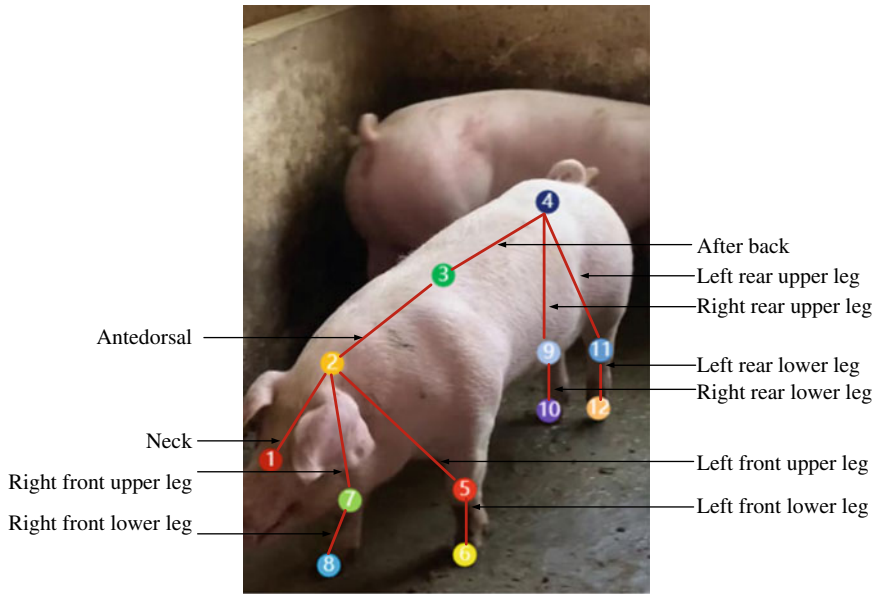11 limbs are defined in this paper, as shown in Fig. 51.6.

Fig. 51.6 Definition of pig limbs

In this paper, there are 22 PAFs output channels, including 11 limb direction vectors. When σ_l set to 0.0015, the key affinity domains of 11 limbs are obtained, as shown in Fig. 51.7.
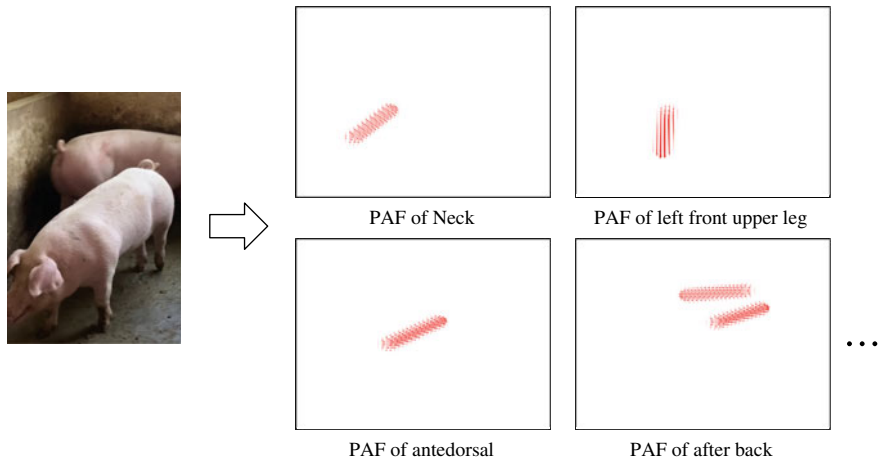

Fig. 51.7 PAFs output. The arrow direction of the output of PAF of the limb represents the direction of vectors in the affinity domain of key points

A set of discrete candidate positions of key points can be obtained from the confidence graph of key points. All candidate key points constitute a set $D_J = \left\{ d_j^m : \text{for } j \in \{1 \dots J\}, m \in \{1 \dots N_j\} \right\}$, define variables $z_{j_1 j_2}^{mn} \in \{0, 1\}$ to indicate candidate key points and whether $d_{j_1}^m$ and $d_{j_2}^n$ can be joined to get the collection. Therefore, get the set $Z = \left\{ z_{j_1 j_2}^{mn} : \text{for } j_1, j_2 \in \{1 \dots J\}, m \in \{1 \dots N_{j_1}\}, n \in \{1 \dots N_{j_2}\} \right\}$ when only the limb c is considered, the two key points corresponding to it are respectively represented by j_1 and j_2, and the candidate set corresponding to the key point is respectively represented by D_{j_1} and D_{j_2}, then the objective function is shown in the formula (51.9),

$$
\begin{aligned}
\max_{Z_c} E_c &= \max_{Z_c} \sum_{m \in D_{j_1}} \sum_{n \in D_{j_2}} E_{mn} \cdot z_{j_1 j_2}^{mn}, \\
s.t. \quad &\forall m \in D_{j_1}, \sum_{n \in D_{j_2}} z_{j_1 j_2}^{mn} \leq 1, \\
&\forall n \in D_{j_2}, \sum_{m \in D_{j_1}} z_{j_1 j_2}^{mn} \leq 1.
\end{aligned}
\tag{51.9}
$$

E_c represents the sum of weights corresponding to the trunk c, Z_c is the subset Z corresponding to the trunk c, and E_{mn} is the affinity of the key point $d_{j_1}^m$ and $d_{j_2}^n$. For all the torsos, the optimization objective function is transformed into the formula (51.10),

$$
\max_{Z} E = \sum_{c=1}^{C} \max_{Z_c} E_c.
\tag{51.10}
$$

51.2.3 Attitude Classification of Pigs

Typical posture of pigs

Four typical pig postures were defined, including crawling, lying on one's side, standing, and sitting. The four attitudes are shown in Table 51.1.

Attitude feature extraction

In this paper, a mixture of joint angles and joint spacing is used to describe the posture of pigs. Take any two segments of bone in an individual pig as an example, the coordinates of the corresponding key points are $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$, $D(x_4, y_4)$, then the joint angle θ can be calculated by formula (51.11), and the joint spacing L can be calculated by formula (51.12),

Table 51.1 Schematic diagram of four kinds of pigs

Type	Schematic diagram
Prone position	
Lateral position	
Standing position	
Sitting-dog position	

$$\cos\theta = \frac{\overrightarrow{AB} \cdot \overrightarrow{CD}}{(|\overrightarrow{AB}| \cdot |\overrightarrow{CD}|)}, \tag{51.11}$$

$$L = \frac{L_{MN}}{d}\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \tag{51.12}$$

L_{MN} is the head height of the individual, and d is the standard head height.

In this paper, 10 groups of joint angle characteristics and 15 groups of joint spacing characteristics were defined. The angle position of each joint was shown in Fig. 51.8, and the distance position of each joint was shown in Fig. 51.9. 10 groups of joint angles.

Pig attitude classification method based on KNN algorithm

When an unknown sample is input, among the K samples closest to the unknown sample, the category that accounts for more determines the category of the unknown

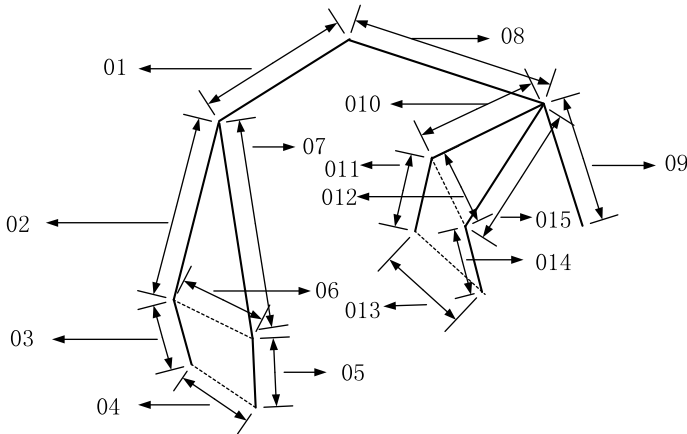**Fig. 51.8** 10 groups of joint angles
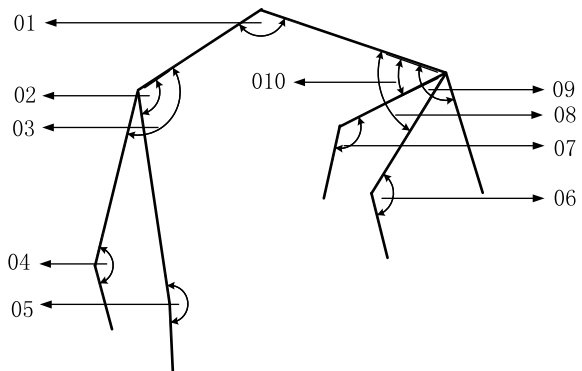
Fig. 51.9 15 groups of joint spacing

sample [5]. In this paper, the Euclidean distance is used to represent the distance between the unknown sample and the sample in the feature space. The Euclidean distance between two points $x_1(x_{11}, x_{12} \ldots, x_{1n})$ and $x_2(x_{21}, x_{22} \ldots, x_{2n})$ in the n dimensional feature space can be calculated by the formula (51.13),

$$L_2 = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2}. \tag{51.13}$$

In the process of experiment, this paper adopts normalization treatment for the values of all sample features and determines the value of K through cross validation. The algorithm process is shown in Fig. 51.10.

51.2.4 Evaluation Indicators

In this paper, the Accuracy (Acc), the Precision (P), the Recall (R), the F_1 score, and Frames per second (FPS) were used as evaluation indexes of the model, which were calculated by formula (51.14),
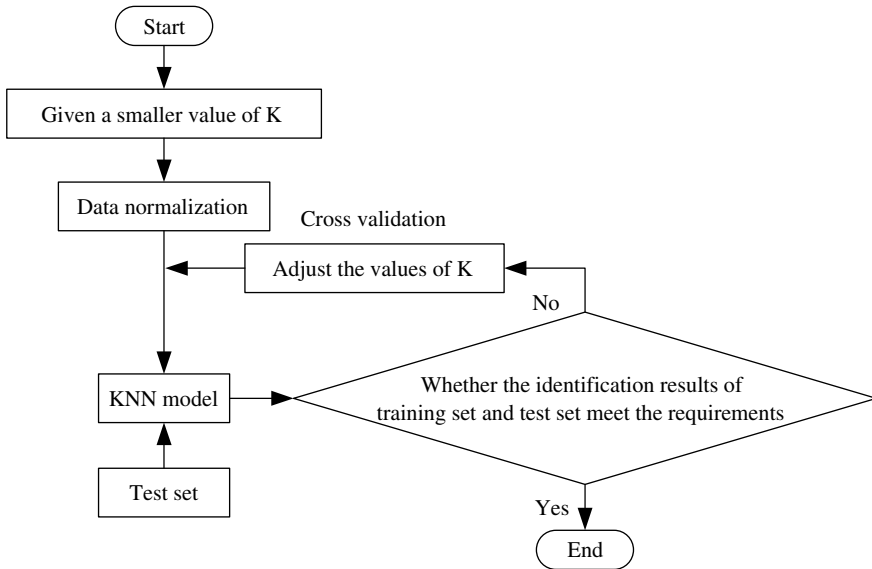
Fig. 51.10 Pose recognition training and testing process based on KNN algorithm

$$
\begin{aligned}
Acc &= \frac{TP+TN}{TP+TN+FP+FN}, \\
P &= \frac{TP}{TP+FP}, \\
R &= \frac{TP}{TP + FN}, \\
F_1 &= 2 \cdot \frac{P \cdot R}{P + R}, \\
FPS &= \frac{frameNum}{elapsedTime}.
\end{aligned}
\tag{51.14}
$$

TP represents the true example, TN represents the true counter example, FP resents the false positive example, FN represents the false counter example, frameNum represents the image frame number, and elpsedTime represents the time interval.

51.3 Results and Discussion

The experimental platform GPU adopts NVIDIA GTX 1080TI, 11 GB video memory, 16G installation memory, TensorFlow version 2.1.0, Python version 3.6 Related experimental parameters are shown in Table 51.2.

The image data of the test set was input into the pose recognition model, and the pose category of pigs was predicted. The recognition results were shown in Table 51.3.

Table 51.2 Experimental parameters

The parameter name	The parameter value
Image size	360×520
Epoch	50
Batch size	10
Learning rate	0.001
σ	0.008
σ_l	0.0015
KNN algorithm initial K value	20

Table 51.3 Attitude recognition results

	P	R	F1	Support
Prone position	0.94	0.92	0.93	912
Lateral position	0.92	0.94	0.93	875
Standing position	0.93	0.92	0.92	922
Sitting-dog position	0.93	0.92	0.92	875
Accuracy	0.93	0.93	0.93	0.93
Macro avg	0.93	0.93	0.93	3584
Weighted avg	0.93	0.93	0.93	3584

It can be seen from Table 51.3 that the average accuracy of pig attitude classification algorithm based on KNN algorithm, which takes joint angle and joint spacing as input features, is more than 93% for pig attitude recognition. Among them, the recognition accuracy of prone position is the highest, reaching 94%, while the recognition accuracy of lateral position is lower, reaching 92%. The reason for the low recognition accuracy of lateral position may be that there are more key points on the body of pigs when they are lying on their side, which leads to less characteristics of input joint angle and joint spacing, and thus reduces the recognition accuracy of posture.

During the experiment, the FPS of the model is stable between 10 and 12, and the real-time performance is normal. It may be because the first ten layers of VGG19 network are used as the feature extractor, which makes the model a large amount of computation and takes a long time. However, through the observation of the daily behavior of pigs, it is found that the posture of pigs is relatively fixed within a period of time. The model can meet the requirements of real-time recognition of pig posture.

51.4 Conclusions

In this paper, a pig pose recognition method based on deep learning is proposed for pig pose recognition. Firstly, the collected images were input into the OpenPose pose estimation model to obtain the coordinate information of the key points and the connection information of the key points. The two groups of information were converted into joint angle features and joint spacing features, which were input into the KNN algorithm to classify the poses of the pigs. The experimental results show that the accuracy of the model for pig attitude recognition is more than 93%, and the FPS is between 10 and 12, which is a method with both accuracy and speed.

However, this method also has some limitations: (1) the data are collected in the same breeding base and the scene is single; (2) the recognition speed is slow. In view of the above problems, the following research will further expand the data set and optimize the attitude estimation model and attitude classification algorithm to improve the detection efficiency under the condition of ensuring the accuracy.

References

1. Q. Yang, D. Xiao, A review of video-based pig behavior recognition. J. Sci. Appl. Animal Behav. Sci. **2020**, 233 (2020)
2. S. Viazzi, G. Ismayilova, M. Oczak et al., Image feature extraction for classification of aggressive interactions among pigs. J. Comput. Electron. Agricult. **104**, 57–62 (2014)
3. L. Jonguk, L. Jin, P. Daihee et al., Automatic recognition of aggressive behavior in pigs using a kinect depth sensor. J. Sensors **16**(5), 631 (2016)
4. C. Zhe, T. Simon, S.E. Wei, et al., Realtime multi-person 2D pose estimation using part affinity fields, in *2017 IEEE Conference on Computer Vision and Pattern Recognition*, (Hawaii, 2017), pp. 172–186
5. T. Abeywickrama, M.A. Cheema, D. Taniar, K-nearest neighbors on road networks: a journey in experimentation and in-memory implementation. J. Proc. VLDB Endowment **9**(6) (2016)
6. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. J. Comput. Sci. (2014)