STUDY ON THE INFLUENCE OF PCA PRE-TREATMENT ON PIG FACE IDENTIFICATION WITH KNN
/
PCA 前处理对 KNN 识别猪脸的影响研究

Hongwen YAN*, Zhiwei HU, Yiran LIU
College of Information Science and Engineering, Shanxi Agricultural University, Taigu/China
Tel: +86-0354-6288165; E-mail: yhwhxh@126.com
DOI: https://doi.org/10.35633/inmateh-70-08

Keywords: *KNN, PCA, individual identification, intelligent management of pig breeding*

ABSTRACT
To explore the application of traditional machine learning model in the intelligent management of pigs, in this paper, the influence of the Principal Component Analysis (PCA for short) pre-treatment on pig face identification with k-Nearest Neighbor (KNN for short) is studied. With testing method, individual identification test was carried out on 10 different pigs in two testing schemes, in which one adopted KNN alone and the other adopted PCA + KNN, for which the classifier parameter was taken as 3 and 5, respectively. In the optimized scheme, the operating efficiency got significantly increased, also the training time and testing time were reduced to 4.8% and 7% of the original value in the KNN alone scheme, though the accuracy got lowered to a certain extent. With all these factors taken into consideration, PCA pre-treatment is beneficial to individual pig identification with KNN. It can provide experimental support for mobile terminals and embedded application of KNN classifiers.

摘要
为探索传统机器学习模型在生猪智能管理中的应用，本文研究了 PCA 前处理对 KNN 识别猪脸的影响，采用试验方式分别确定仅采用 KNN 以及 PCA+KNN 两种试验方案分类器参数值分别为 3、5，分别对 10 头生猪进行个体识别试验，优化方案运行效率显著提升，训练时间和测试时间缩减为原来的 4.8%、7%，准确率有一定程度降低，综合考虑，使用 PCA 前处理对采用 KNN 进行生猪个体识别具有增益作用，可为 KNN 分类器的移动端和嵌入式应用提供试验支持。

INTRODUCTION
The management of individual identification and behavior analysis of pigs, an important part in the intelligent management of pigs, can be divided into three categories: the first category is based mainly on RFID (Radio Frequency Identification) technology *(Maselyne et al., 2014; Hahnel et al., 2016)*, while the second category is based mainly on traditional machine learning model: LASSO regression and random forest model were used to predict the weight of pigs at 159 days to 166 days under four scenarios *(He et al., 2021)*; random forest and generalized linear regression were used to predict physiological temperature of piglets, though the prediction error was relatively high *(Gorczyca et al., 2018)*; auto-regression (AR) model and improved local linear embedding (LLE) were used to estimate pig weight in actual farm environment *(Wongsriworaphon et al., 2015)*. The third category is based mainly on the application of deep learning model and improved computer vision technology. Tu Shuqing et al. explored a PigMS R-CNN (Region Convolutional Neural Networks) framework based on mask scoring R-CNN (MS R-CNN) to segment the adhesion regions in images for herd pigs as well as identify and locate them. Zhang Jianlong et al. modified DenseNet201, ResNet152 V2, Xception and MobileNet V2 to be a multi-output regression CNN (Convolutional Neural Network) before getting them trained on modeling data, with modified Xception selected as the optimal estimation model. In order to improve the real-time performance of the model, Residual learning structure was introduced in, with its MSE reaching 0.092 *(Zhang et al., 2021)*. Yan Hongwen used the Feature Pyramid Attention (FPA) combined with a Tiny-YOLO model to achieve multi-target detection of herd pigs in different scenarios *(Yan et al., 2020)*. Yan Hongwen realized the detection of different types of face posture of herd pigs by constructing an attention sub-model with which both spatial attention and channel attention information are respectively introduced in on basis of YOLOV3 model *(Yan et al., 2019)*.

[1] *Hongwen Yan*, As. Ph.D. Prof. Eng.; Zhiwei Hu, Lect. Ph.D. Eng.; Yiran Liu, Lect.. Ph.D. Eng.*

Hu Zhiwei constructed a Dual Attention Unit (DAU) which integrates Channel Attention Unit (CAU) and Position Attention Unit (PAU) as one by taking ResNet50 and ResNet101 as the backbone network, and had it used in the Feature Pyramid Network (FPN) structure to realize the pig instance detection in different scenarios (*Hu et al., 2021)*. Marsot *(Marsot et al., 2020)* used the cascade classifier with Haar features to intuitively see how neural network learns to distinguish parameters by using the class activation map generated by grad-CAM, with an accuracy for 320 test images reaching 83%.

Though with the application of GPU, the high precision advantage of computer vision technology has gradually been revealed, in the management of small and medium-sized farms, the needs for mobile terminals and embedded application have gradually been increasing. For the deep learning model has high requirements to hardware, it is difficult to adapt to wide application. In addition, the deep learning model is of weak interpretability, which leads to its weak controllability. That is why it is necessary to study the application of RFID and traditional machine learning model in the management of small and medium-sized farms. However, for the RFID technology is easy to be simulated in the management *(Shi et al., 2012; Zhou et al., 2018)*, also it violates the animal welfare breeding thinking, it has gradually been eliminated. Only the requirements to hardware by traditional machine learning model conform to the standards for mobile terminals and embedded application, though there is still room for improvement in its identification accuracy and running time. With principal component analysis, the main features of the images for pig faces can be extracted *(Lin et al., 2021; Zhang et al., 2021)*, thus reducing the computation burden, improving the interpretability of the model, the operating efficiency, eliminating noise interference and improving identification accuracy. In order to promote the use of traditional machine learning model in mobile terminals and embedded application, in this paper, KNN is adopted for bettering the identification efficiency of pig faces in this study, and its influence on the efficiency of pig face identification is further studied by adding PCA pre-treatment in, which provides experimental support for its use in both mobile terminals and embedded application.

MATERIALS AND METHODS

Sample collection

The data of this study was collected in two times. The first time was collected in Dongsongjiazhuang Village, Jicun Town, Fenyang City, Shanxi Province, China (111º95′ E, 37º27′ N). In order to obtain live pig images of different pig house scenes, it was collected from 9:00-14:00 on June 1, 2019 (fine, strong light). Select 3 pig farms for video capture, each pig farm consists of 10-30 pig pens, the number of each pen varies from 6 to 8, the size of pig pens is about 3.5 m×2.5 m×1 m. A total of 35 videos of 5 pens of breeding pigs aged 20 to 105 days were collected. The second time was from 10:30-12:00 on October 13, 2019 (cloudy, weak light), and the collection site was located in the Laboratory Animal Management Center of Shanxi Agricultural University, Taigu City, Shanxi Province, China (112º59′ E, 37º43′ N). A total of 15 pigs in 6 pens are selected for video collection. In this study, 10 pigs are selected as the research objects, as shown in figure 1, including 768 training samples, 85 validation samples and 250 test samples.

Fig. 1 - Pig Samples

The computer used in the experiment is configured with 64-bit windows system, Intel Core i7-6700, 8GB memory, 6GB video memory capacity, and Program development uses Python V3.5 version language.

Principle of pig face identification with KNN

Among machine learning methods, the nearest neighbor algorithm can be traced back to the study made by Fix and Hodges *(Evelyn et al., 1989)*, and a more detailed content on k-nearest neighbor algorithm was proposed by Dasarathy *(Dasarathy et al., 1991)*. The k-nearest neighbor algorithm aiming to measure the distance between different eigenvalues for classification is also called KNN algorithm. In principle, KNN works in this way: the labeled sample set is used to figure out the data samples of the nearest distance by calculating the features of new data and the corresponding features in the training data, then the category of the samples with the nearest distance is judged before the category of new data is output. In KNN algorithm, k samples with the nearest distance can be chosen, therefore, before using the KNN algorithm to analyze the data, we need to determine the value of the parameter k first. In this study, the KNN algorithm was used to analyze pig face data in the following process:

(1) Data collection. Collect data to obtain image data of pig face;

(2) Data preparation. Prepare the numerical value needed for distance calculation, then normalize the pig face image data, and represent the matrix representing each image as a vector, so as to facilitate the calculation of distances between sample data;

(3) Data analysis. Divide the data into training data and test data by either sampling or proportion segmentation method;

(4) Algorithm training. KNN algorithm is used to train the training sample set of pig face data, in which k is an independent variable, and the KNN is trained by inputting different k values;

(5) Algorithm testing. Calculate the error rate of KNN algorithm for k of different values, so as to determine the optimal k value;

(6) Category determination. Use the algorithm to input the determined optimal k value into the KNN algorithm model, and then run the KNN algorithm on the test sample set of pig face data to determine which category the input pig face data for test respectively belong to.

In KNN, instance-level samples are used for training, so all the data need to be stored in the training process. Usually, a large number of training samples are used, and the corresponding storage cost is relatively high. When the algorithm is running, the distance of each sample needs to be calculated, which causes a certain time cost, for which Euclidean distance is used. The L_p distance is defined in the following way:

$$L_p(x_i, x_j) = \left(\sum_{i=1}^{n} \left|x_i^{(l)} - x_j^{(l)}\right|^p\right)^{\frac{1}{p}} \tag{1}$$

where: $x_i \in R^n$ represents a point in the dataset to be used for classification reference, [dimensionless];

$x_j \in R^n$ represents another point in the dataset to be used for classification reference, [dimensionless];

L_p represents the category parameter of the calculated distance in the KNN algorithm, [dimensionless];

l represents the dimension of the data point, [dimensionless].

p is a variable element.

In this study, p is taken as 2, namely Euclidean distance, corresponding to L_2 norm.

According to the given distance, we can find the k points closest to the data point x in the training set, and the field covering the k points is denoted as $N_k(x)$. In $N_k(x)$, according to the classification decision rules, the algorithm can decide the class of the data point x, which is represented by y, the y is defined in the following way:

$$y = argmax_{c_j} \sum_{c_j \in N_k(x)} I(y_i = c_j) \tag{2}$$

where: I represents the indicator function, when $y_i = c_j$, I is 1, otherwise I is 0, [dimensionless];

$N_k(x)$ represents the area of k points closest to the data point x, [dimensionless].

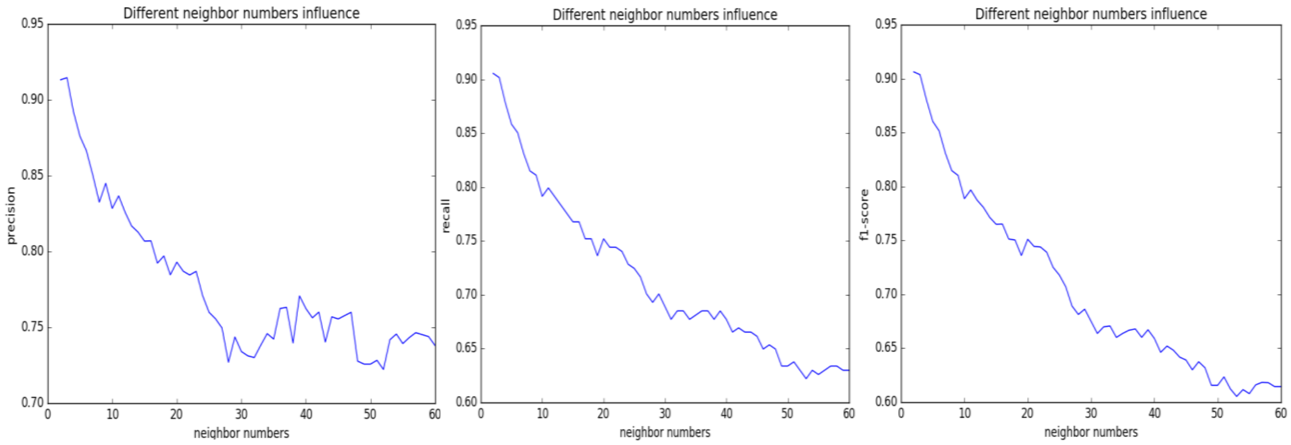
RESULTS AND DISCUSSION

Pig face identification test carried out with KNN alone

KNN model parameter determination

In the feature space, k features can always be found to have certain similarity with an input sample, which can be measured by the Euclidean distance between vectors. If most of the k samples belong to a certain category, then the sample also belongs to this category, where k is usually an integer not greater than 20. KNN algorithm determines which category the samples to be detected should be divided into according to the category of the selected k samples. Therefore, the selection of k is a key factor for the performance of KNN algorithm.

There is no fixed method for the selection of k value. In accordance with the different distribution of samples, a small value is first selected to test the indicator change of classifier, and k value with better comprehensive effect is selected. In order to determine the value of k, in this paper, it was taken as an independent variable to study the relationship between different k values and KNN's classification accuracy, recall rate as well as *f1* value of original image data. The curve relationship drawn is as shown in figure 2.



a. accuracy *b. recall rate* *c. f1 value*

Fig. 2 - The relationship between the evaluation index of KNN model and its kernel function and coefficient

Fig. 2(a) shows the relationship between the k value and the accuracy rate, wherein the abscissa represents k value while the ordinate represents the accuracy rate. As can be seen from the figure, when k value was less than 25, the smaller the k value was, the higher the accuracy rate was; when k value was greater than 25, the accuracy rate was low and showed fluctuation; when k value was greater than 55, the accuracy rate was basically smooth and steady. This phenomenon appeared for the reason that in KNN algorithm, the category of the unknown pig face data to be predicted is determined by the label of k data that are nearest to the unknown data. Therefore, the larger the k value is, the more the categories possibly contained in the k comparative values selected are. When the predicted value is equal to the one with the most labels among k samples, the accuracy may decline.

Fig. 2(b) shows the relationship between the k value and the recall rate, wherein the abscissa represents k value and the ordinate represents the recall rate. Different from the results in figure 2(a), the recall rate basically showed a decreasing trend as k value increased. However, when k reached a certain value, the recall rate was close to a fixed value.

Fig. 2(c) shows the relationship between the k value and the *f1* value, wherein the abscissa represents the k value and the ordinate represents the *f1* value. Similar to the results in figure 2(b), as k value changed, the change of *f1* value was basically consistent with the change of recall rate. As can be seen from the figure, different indicators showed different change characteristics under different k values. As for the accuracy index, its value always decreased in the first half stage when k value increased, but when k increased to a certain extent, there was a great fluctuation, while the recall rate and *f1* value basically decreased as k value continued to increase. With a comprehensive consideration of the performance of the three indicators, in this paper, the k value was taken as 3 to be the parameter for the subsequent test set in KNN model.

Evaluation index of KNN model

According to the parameter determined by the above tests, that is, the k value set to be 3 in KNN for the tests to be carried out on the test set, the prediction confusion matrix was drawn according to the test results, as shown in figure 3. The main diagonal element of the matrix represents the sample size of pig face correctly classified. The background color of each data in the matrix changed from light to dark as the value increased, and the darker the color was, the larger the value was. It can be seen from figure 3 that the KNN algorithm achieved good results in the classification of 10 pigs.

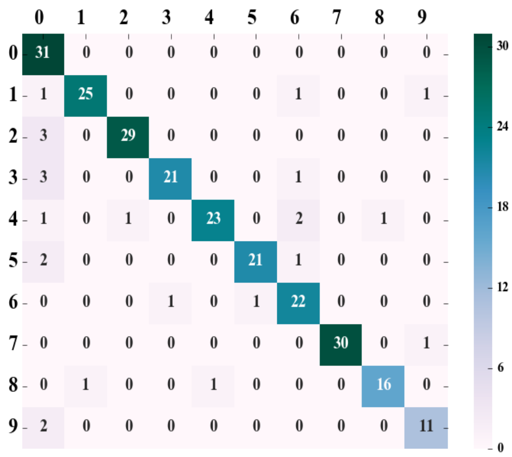


Fig. 3 - KNN prediction result confusion matrix

The prediction accuracy, recall rate and f1 value of each category are calculated in accordance with the confusion matrix, and the calculation formulas are as shown in formula (3) ~ (5). In the classification task, the results were usually divided into true positive cases (TP), false positive cases (FP), true negative cases (TN) and false negative cases (FN). With the number of samples corresponding to TP, FP, TN and FN given, the precision ratio was defined as:

$$precision = \frac{TP}{TP + FP} \tag{3}$$

Recall ratio was defined as:

$$recall = \frac{TP}{TP + FN} \tag{4}$$

Recall ratio was also called recall rate. Recall ratio and precision ratio changed in opposite trend. $f1$-score can measure the different preferences of these two indexes, and the formula was as follows

$$f1 - score = 2 \times \frac{precision \times recall}{precision + recall} \tag{5}$$

where: TP represents the number of positive samples that are actually positive samples, [a];
FP represents the number of positive samples that are actually negative samples, [a];
FN represents the number of negative samples that are actually positive samples, [a].

Table 1

KNN model prediction performance table

Category	*precision* [%]	*recall* [%]	*f1-score* [%]	Count [a]
1	72	100	84	31
2	96	89	93	28
3	97	91	94	32
4	95	84	89	25
5	96	82	88	28
6	95	88	91	24
7	81	92	86	24
8	100	97	98	31
9	94	89	91	18
10	85	85	85	13
average	91.46	90.16	90.36	25

It can be seen from table 1 that the average accuracy rate, recall rate and *f1* value of KNN algorithm for pig face data classification and recognition were up to 91.46%, 90.16% and 90.36% respectively. The effects were good.

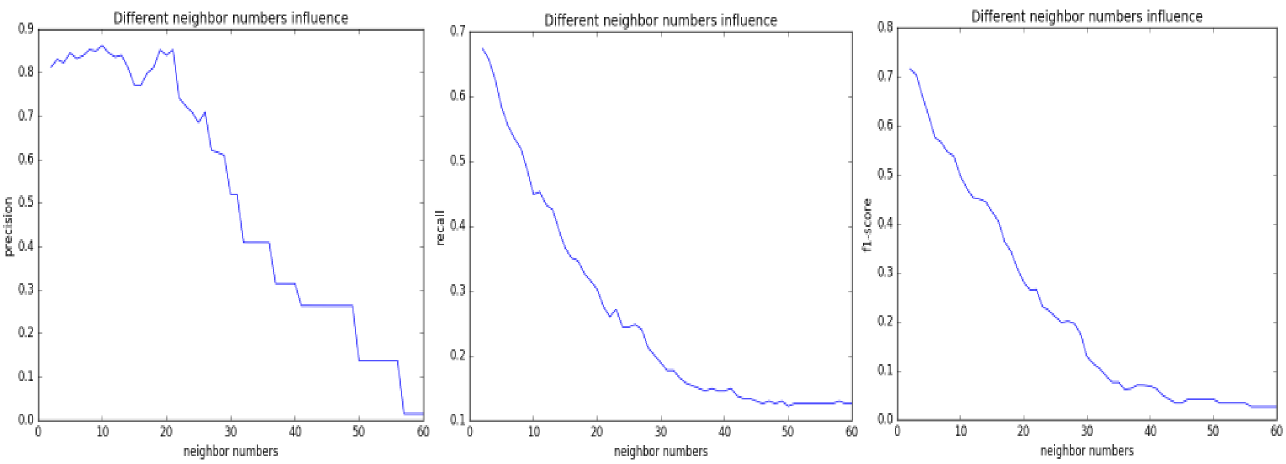
Experiment of pig face identification with KNN + PCA pre-treatment

Determine the *k* value in principal component analysis

At the first stage of the experiment, the number of principal components needs to be determined for principal component analysis. Here the *k* value was taken as 300, with the variance explanation rate reaching over 95%.

Determination of KNN parameters in the optimization plan

Distribution of the data going through PCA dimension reduction may change to some extent, thus the *k* value determined in the previous stage may not be the optimal value when it is used to be the input for KNN algorithm. Therefore, the *k* value of KNN algorithm needs to be redetermined. With the testing method in "**KNN model parameter determination**" adopted, the relationships between different *k* values and the classification accuracy, recall rate as well as *f1* value of KNN to pig face data were measured, and the relationship curves are as shown in figure 4.



a. accuracy *b. recall rate* *c. f1 value*

Fig. 4 - The relationship between KNN model performance and parameters after PCA pre-processing

In figure 4(a), the abscissa represents *k* value, and the ordinate represents precision. When *k* is 0, the image to be tested will not be predicted to belong to any category.

When *k* is the maximum, the precision value is close to 0, for which the reason is that any image to be tested will be predicted to belong to the category with the largest number of samples in the data set. Obviously, in this way, the accuracy would be very low. It can be seen from the figure that the *k* value began with 2, and as *k* value increased, the accuracy fluctuated greatly. As *k* value increased within the range of 2~12, the precision value gradually increased, reaching 0.85. This is because a small *k* value makes the training error very small. The prediction can be correct only when the sample to be tested has the same or similar distribution with the training sample. When the generalization error of the classifier increases, the classification model for training will be very complex and over-fitting may occur, so at this stage, the precision increases as *k* value increases.

When *k* value exceeded 12, its corresponding precision value started to decrease and reached a stage lowest value of 0.77 when it reached 14. However, the precision value did not always decrease but rebounded and reached the highest value of 0.86 when *k* value reached 20. This indicates that the precision value decreases for the over-fitting situation caused by noise interference of the sample, thus the precision value may decrease for a while. As *k* value increases, the training data are enlarged, partly solving the over-fitting problem. Therefore, the precision value rises again.

When *k* value exceeded 20, its corresponding precision value was always in a declining trend, though rising slightly at 28, while falling continuously after 28. At the ranges of 30~35, 40~48, 50~57 and 58~60, several smooth and steady curves could be seen. At these smooth and steady stages, the highest precision was 0.4 and the lowest was about 0.01, which means that such predictions were meaningless.

In figure 4(b), the abscissa represents k value, and the ordinate represents recall. While precision deals with accuracy, recall solves the problem of incompleteness. An ideal model requires these two values to be high. When recall is 0, it means that all positive examples are not predicted. When recall value is 1, it indicates that all positive instances can be predicted, which is considered to be the most ideal model.

As can be seen from figure 4(b), when k value was 2, its recall value reached the peak, and then it kept declining. When k value was 10, it stayed flat slightly before continuing to decline. When k value was 21, it rose slightly, and then kept declining. The whole declining process gradually got flat and remained unchanged after k value turned 50. When k was 2, recall reached the maximum value of 0.68. According to the probability theory, the more the sample points are, the more categories it is likely to contain; conversely, the fewer the sample points are, the fewer categories it is likely to contain. According to KNN principle, when k value is small, the more probable it can correctly predict positive samples, if stated in the figure, recall will be at a high position. As k value increased, the probability of correctly predicting positive samples got gradually decreasing, and its curve was in a downward trend.

As can be seen from figure 4(a), the precision value was obviously interrupted by noise when k value reached 14. It was not obvious in recall curve showing only being slightly flat. The prediction error rate gradually decreased as k value increased, that is why the curve gradually turned placid. When k exceeded 30, its precision was lower than 0.5, which means that such prediction was meaningless. When k exceeded 50, the prediction was basically wrong, and the curve was basically in a horizontal state with a very small value, which was about 0.12 as in the figure. Considering precision and recall values comprehensively, k should be within 2~8. In this range, it can be ensured that the recall value was greater than 0.5 and the precision value was greater than 0.8 and in an upward trend.

For *f1*, the classification accuracy rate and recall rate needs to be considered. With a value range of 0~1, *f1* score assumes that accuracy and recall rate are equally important, so the one with a bigger value shall be chosen. In figure 4 (c), it can be seen that when the k value was 2, *f1* score was 0.72. As k increased, *f1* score was in decline, though it rose slightly in the middle. After k exceeded 40, the *f1* value remained basically at about 0.02. With all the three factors considered, going through tests on the training set, the k value of KNN was finally selected to be 5 before it got tested on the validation set.

Model evaluation index of optimization plan

The principal component value of principal component analysis measured on the training set was selected to be 300, and the k value of KNN classifier was selected to be 5, and they were tested in the test set. With the test results, the confusion matrix was drawn, as shown in figure 5.

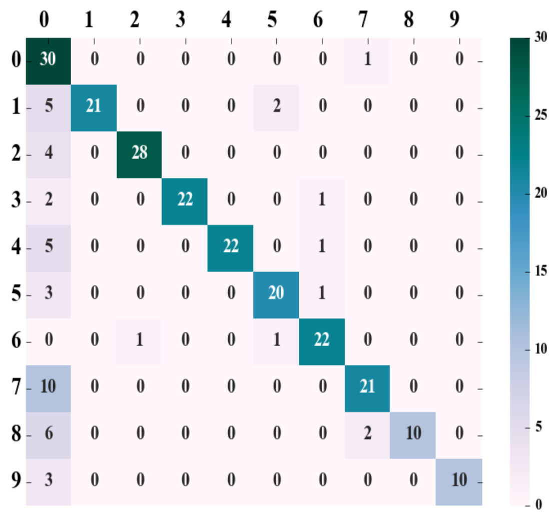


Fig. 5 - PCA+KNN prediction result confusion matrix

The precision, recall and *f1*-core values of 10 different pigs were obtained in accordance with the confusion matrix and formulas (6) ~ (8), as shown in table 2.

Table2

PCA+KNN model prediction performance table

Category	Precision [%]	Recall [%]	*f*1-score [%]	Count [a]
1	25	100	40	31
2	93	50	65	28
3	78	66	71	32
4	100	52	68	25
5	100	36	53	28
6	67	50	57	24
7	100	54	70	24
8	92	39	55	31
9	100	56	71	18
10	92	85	88	13
average	82.82	57.87	61.86	25

As can be seen from table 2, the average precision of PCA+KNN classifier for 10 classified pig samples reached 82.82% after feature extraction by principal component analysis. With individual pig identification with KNN + PCA pre-treatment, the identification accuracy got reduced to a certain extent, though the training time and test time were greatly improved. The specific test indexes of the two schemes are shown in table 3.

Table3

The optimization result of KNN model by PCA pre-processing

Model	Precision [%]	Precision change	Test_time [ms]	Test_ new / old [%]	Train_time [ms]	Traintest_ new / old [%]
KNN	91.46		1,306		187	
PCA+KNN	82.82	-8.64	93	7	9	4.8

As can be seen from table 3, in the test, in the PCA+KNN test scheme, the training time and the recognition time were 9 ms and 93 ms respectively, while the precision, the training time and the identification time of KNN classifier without feature extraction by principal component analysis were 91.46%, 187 ms, and 1,306 ms respectively, indicating a decrease in the identification accuracy and a great rise in operating speed. The training time was 4.8% of the original value in the KNN alone scheme, while the identification time was 7% of the original value.

PCA+KNN test results showed that the precision decreased from 91.46% to 82.82%, which may be caused by the difference in data distribution between the test set and the training set. The much noise in pig samples in the test set was also one of the reasons for the reduced accuracy. The operating efficiency got significantly improved: the training time and the identification time were reduced to 4.8% and 7% of the original values respectively, much better than the expected test effects. Probably the reason is that going through the PCA feature extraction, on the one hand, pig facial features for calculation got reduced, thus reducing the amount of calculation and enhancing the efficiency of the algorithm; on the other hand, some minor features were eliminated while the main features were extracted, thus the image noise was eliminated and the noise reduction effect was achieved. Theoretically, it is predicted that both the recognition accuracy and the operating efficiency of the algorithm would be improved.

The experimental results showed that the precision was reduced to a certain extent, which was different from the expected effect of the test. It could be adjusted in the distribution of sample types, but the accuracy still remained at 82.82%. With the operating efficiency of the algorithm greatly improved, the loss of accuracy was within an acceptable range.

CONCLUSIONS

In this paper, the influence of PCA pre-treatment mode on the efficiency of identification of 10 different pigs with a KNN classifier employed was studied. The parameter for the classifiers was determined via tests. With the influence of two test methods on identification efficiency compared, in which one adopting the KNN classifier alone and the other adopting PCA+KNN, we got the following conclusions:

(1) When KNN classifier is used for individual pig identification, the selection of kernel function is related to the pre-treatment method. When the KNN alone scheme is adopted, the parameter value can be 3; while the PCA+KNN scheme is adopted, the parameter value can be 5;

(2) PCA pre-treatment can benefit the efficiency of individual pig identification with KNN adopted, with the accuracy decreasing to 82.82% from 91.46%, and the training time as well as testing time reduced to 4.8% and 7% of the original value respectively;

(3) KNN classifier going through PCA pre-treatment is more suitable for the application of mobile terminals and embedded devices, so it suits to the development of a portable and real-time pig face identification system.

ACKNOWLEDGEMENTS

This research, titled ‘Study on the Influence of PCA Pre-treatment on Pig Face Identification with KNN’, was funded by the National Key Research and Development Plan of China (2016YFD0701801), the Shanxi Province Basic Research Program Project (Free Exploration) (Grant No.20210302124523, 202103021224149, 202103021223141), the Doctor Scientific Research Foundation of Shanxi Agricultural University (2020BQ14). The authors are grateful and honored to have obtained support from the Key Laboratory of Biomechanics.

REFERENCES

[1] Adrion F, Kapun A, Holland E, et al., (2017), Novel approach to determine the influence of pig and cattle ears on the performance of passive UHF-RFID ear tags, *Computers and Electronics in Agriculture*, Vol.140, pp.168–179, Oxford/England;

[2] Dasarathy, B.V., (1991), *Nearest neighbor (NN) norms: NN pattern classification techniques*, IEEE Computer Society, pp.5-20, Hoboken/ USA;

[3] Fix E., Hodges J.L., (1989), Discriminatory analysis-nonparametric discrimination: consistency properties, *Revue Internationale de Statistique*, Vol.57, Issue 3, pp.238-247, California/USA;

[4] Gorczyca M. T., Hugo F. M., Alex S. C., et al., (2018), Machine learning algorithms to predict core, skin, and hair-coat temperatures of piglets*, Computers and Electronics in Agriculture,* Vol.151, pp.286–294, Oxford/England;

[5] Hahnel D, Burgard W, Fox D, et al., (2016), Mapping and localization with RFID technology. *IEEE International Conference on Robotics and Automation*, pp.1015-1020, Los Angeles/USA;

[6] He Y.Q., Tiezzi F., Jeremy H., et al., (2021), Predicting body weight in growing pigs from feeding behavior data using machine learning algorithms, *Computers and Electronics in Agriculture*, Vol.184, pp.106085, Oxford/England;

[7] Hu Z.W., Yang H., Lou T.T., (2021), Dual attention-guided feature pyramid network for instance segmentation of group pigs, *Computers and Electronics in Agriculture*, Vol.186, pp.106140, Oxford/England;

[8] Jarissa M., Wouter S., Petra B., et al., (2016), Methods to construct feeding visits from RFID registrations of growing-finishing pigs at the feed trough, *Computers and Electronics in Agriculture*, Vol.128, pp.9–19, Oxford/England;

[9] Lin H. , Sheng H. , Sun G.X. , et al., (2021), Identification of pumpkin powdery mildew based on image processing PCA and machine learning, *Multimedia Tools Applications*, Vol.80, Issue 14, pp.21085-21099, Dordrecht / Netherlands;

[10] Marsot M., Mei J.Q., Shan X.C., et al., (2020), An adaptive pig face recognition approach using Convolutional Neural Networks. *Computers and Electronics in Agriculture*, Vol.173 pp.105386, Oxford/England;

[11] Maselyne J, Saeys W, Ketelaere B, et al., (2014), Validation of a High Frequency Radio Frequency Identification (HF RFID) system for registering feeding patterns of growing-finishing pigs. *Computers and Electronics in Agriculture*, Vol.102 pp.10-18, Elsevier, Oxford/England;

[12] Shi Y.W., Zhang Y.Q., Liu K.S., (2012), Research on Security Issues Based on RFID System, *Computer Science*, Vol.39, Issue 6A, pp.214-216, Chongqing/China;

[13] Tu S., Yuan W., Liang Y., Wang F., Wan H., (2021), Automatic Detection and Segmentation for Group-Housed Pigs Based on PigMS R-CNN, *Sensors,* Vol.21, Issue 9, pp.3251, Basel/ Switzerland;

[14] Wongsriworaphon A., Arnonkijpanich B., Pathumnakul S., (2015), An approach based on digital image analysis to estimate the live weights of pigs in farm environments*, Computers and Electronics in Agriculture*, Vol.115, pp.26–33, Oxford/England;

[15] Yan H.W., Liu Z.Y., Cui Q.L., et al., (2020), Multi-target Pig Detection Based on Feature Pyramid Attention and Deep Convolutional Network*, Transactions of the Chinese Society of Agricultural Engineering,* Vol.36, Issue 11, pp.193–202, Beijing/China;

[16] Yan H.W., Liu Z.Y., Cui Q.L., et al., (2019), Detection of facial gestures of group pigs based on improved Tiny-YOL, *Transactions of the Chinese Society of Agricultural Engineering,* Vol.35, Issue 18, pp.169–179, Beijing/China;

[17] Yan H.W., Cui Q.L., Liu Z.Y., (2020), Pig Face Identification Based on Improved AlexNet Model, *INMATEH-Agricultural Engineering*, Vol.61, Issue 2, pp.97-104, Bucharest/Romania; https://doi.org/10.35633/inmateh-61-11

[18] Zhang J.L., Zhuang Y.R., Ji H.Y., Teng G.H., (2021), Pig Weight and Body Size Estimation Using a Multiple Output Regression Convolutional Neural Network: A Fast and Fully Automatic Method, *Sensors*, Vol.21, Issue 9, pp.3218, Basel/ Switzerland;

[19] Zhang Y.Z.*,* Guo Y.N.*,* (2021), Train Driver State Detection System Based on PCA and SVM, *Artificial Intelligence and Security - 7th International Conference*, Vol.12736, pp.516-526, Dublin/Ireland;

[20] Zhou J., Lu J.Z., Tu Y.Z., (2018), Discussion on the use of intelligent recognition terminal to promote the management of animal ear tags, *The Chinese Livestock and Poultry Breeding*, Vol,14, Issue 5, pp.33-34, Beijing/China.