



# Article Automated Behavior Recognition and Tracking of Group-Housed Pigs with an Improved DeepSORT Method

Shuqin Tu <sup>1,2</sup>, Qiantao Zeng <sup>1</sup>, Yun Liang <sup>1,2,\*</sup>, Xiaolong Liu <sup>1</sup>, Lei Huang <sup>1</sup>, Shitong Weng <sup>1</sup> and Qiong Huang <sup>1,2</sup>

- <sup>1</sup> College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China
- <sup>2</sup> Guangzhou Key Laboratory of Intelligent Agriculture, South China Agricultural University,
  - Guangzhou 510642, China Correspondence: yliang@scau.edu.cn

Abstract: Pig behavior recognition and tracking in group-housed livestock are effective aids for health and welfare monitoring in commercial settings. However, due to demanding farm conditions, the targets in the pig videos are heavily occluded and overlapped, and there are illumination changes, which cause error switches of pig identify (ID) in the tracking process and decrease the tracking quality. To solve these problems, this study proposed an improved DeepSORT algorithm for object tracking, which contained three processes. Firstly, two detectors, YOLOX-S and YOLO v5s, were developed to detect pig targets and classify four types of pig behaviors including lying, eating, standing, and other. Then, the improved DeepSORT was developed for pig behavior tracking and reducing error changes of pig ID by improving trajectory processing and data association. Finally, we established the public dataset annotation of group-housed pigs, with 3600 images in a total from 12 videos, which were suitable for pig tracking applications. The advantage of our method includes two aspects. One is that the trajectory processing and data association are improved by aiming at pig-specific scenarios, which are indoor scenes, and the number of pig target objects is stable. This improvement reduces the error switches of pig ID and enhances the stability of the tracking. The other is that the behavior classification information from the detectors is introduced into the tracking algorithm for behavior tracking. In the experiments of pig detection and behavior recognition, the YOLO v5s and YOLOX-S detectors achieved a high precision rate of 99.4% and 98.43%, a recall rate of 99% and 99.23, and a mean average precision (mAP) rate of 99.50% and 99.23%, respectively, with an AP.5:.95 of 89.3% and 87%. In the experiments of pig behavior tracking, the improved DeepSORT algorithm based on YOLOX-S obtained multi-object tracking accuracy (MOTA), ID switches (IDs), and IDF1 of 98.6%,15, and 95.7%, respectively. Compared with DeepSORT, it improved by 1.8% and 6.8% in MOTA and IDF1, respectively, and IDs had a significant decrease, with a decline of 80%. These experiments demonstrate that the improved DeepSORT can achieve pig behavior tracking with stable ID values under commercial conditions and provide scalable technical support for contactless automated pig monitoring.

Keywords: YOLOX-S; pig behavior tracking; YOLO v5s; DeepSORT; ID switch

#### 

**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).

# 1. Introduction

The pig industry has always been the pillar of China's livestock industry. Pig and pork products play a key role in food security and nutrition strategies around the world. The health situation of pigs determines the development and economic benefits of pig farming [1], and the clinical or subclinical signs of most pig diseases are often associated with abnormal pig behavior before the diseases are found. Therefore, the monitoring and analysis of pig activity, diet, and other behaviors can help to quickly understand the health condition of pigs [2,3]. Currently, the manual monitoring pig behaviors is a management method for pig farms, and it requires a lot of labor and is time-consuming and inefficient.



Citation: Tu, S.; Zeng, Q.; Liang, Y.; Liu, X.; Huang, L.; Weng, S.; Huang, Q. Automated Behavior Recognition and Tracking of Group-Housed Pigs with an Improved DeepSORT Method. *Agriculture* **2022**, *12*, 1907. https://doi.org/10.3390/ agriculture12111907

Academic Editor: Wei Ji

Received: 18 October 2022 Accepted: 10 November 2022 Published: 12 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Another method of monitoring pig behavior by wearable smart devices, such as ear tags using RFID wireless radio frequency technology, which can avoid the defects of manual monitoring, usually irritate the pig and reduce the welfare of the pig [4]. In addition, information technology tools have become an important method for the development of modern agriculture. For the past few years, deep-learning-based techniques have been adopted to achieve excellent performance in many fields, such as image recognition [5,6], natural language process [7,8], and so on [9,10]. Thus, it is significant to detect abnormal pig behavior quickly and accurately using computer vision technology at a low cost and no contact to achieve intelligent pig farming [11].

Research has already begun to use automated surveillance techniques to observe pig behavior for the early detection of potential health or welfare problems, such as the a pig's daily behavior [12] and eating and drinking behavior [13,14]. This suggests the feasibility of using computer vision technology for the daily differential behavior monitoring of pigs. Some researchers have implemented livestock behavior monitoring through MOT techniques. For example, a CNN method was developed to classify different types of social behaviors among preweaning piglets [15]. Two deep-learning-based detectors combined with tracking processes were developed to identify pig postures and drinking behaviors of group-housed pigs [16]. A probabilistic tracking-by-detection method was proposed, which first used a fully convolutional detector to detect visible key points of individual pigs and then tracked individual animals in a group setting [17]. A deep-learning-based pig posture and tracking algorithm were designed to measure those behavior changes in an experimental pig barn at different greenhouse gas (GHG) levels [18]. However, pig behavior detection and tracking still face some difficulties, such as target occlusion, varying light, overlapping, and error IDs of tracks in tracking [19,20]. Therefore, to improve the detector and tracker performance, advanced detectors and MOT methods are being introduced to pig behavior tracking applications.

In terms of target detection and classification, anchor-based and anchor-free detectors are widely used for object detection and classification applications. The anchor-based detectors are partitioned into one-stage and two-stage categories, and anchor-free detectors have developed rapidly in the past two years. One-stage algorithms are dominated by the YOLO series [21]. Two-stage algorithms, such as Faster R-CNN [22], are popular. Considering the optimal speed and accuracy trade-off, this study chose the YOLO v5s algorithm as one of the target detectors. The anchor-free detectors include the CornerNet [23], CenterNet [24], and YOLOX [25] algorithms, and so on. Among them, YOLOX adopted advanced detection techniques, i.e., decoupled head and strong data augmentation to achieve state-of-the-art results and won 1st place in the Streaming Perception Challenge (Workshop on Autonomous Driving at CVPR 2021) using a single YOLO-L model; therefore, YOLOX-S was chosen as another of the target detectors. In addition, this study compared the performances of YOLO v5s and YOLOX-S on pig detection and behavior classification.

In terms of MOT, MOT has been a longstanding goal in computer vision. The task of MOT is largely divided into locating multiple objects, maintaining their identities, and yielding their trajectories given an input video [26]. Most existing MOT frameworks can be grouped into three sets, Tracking By Detection (TBD), Joint Detection Embedding (JDE) [27], and MOT based on the attention network [28]. Among them, TBD first adopted the detector to output the detection result and then used the Kalman filter and Hungarian algorithm to accomplish target tracking tasks, such as SORT [29] and DeepSORT [30]. The performance of TBD highly depends on the performance of the employed object detector. The JDE framework developed appearance representation in detection mechanisms are TransTrack [28] and TrackFormer [31], both of which are efforts to apply a Transformer to the MOT method. With significant improvements in the performance of target detectors based on deep learning, DeepSORT has become a simple and efficient MOT method with high accuracy at high speeds. In addition, Deep SORT was found to have a runtime speed

of 25–50 FPS using modern GPUs [30]. Thus, DeepSORT was considered as our basic tracking model.

Accurately tracking individual pigs in real farming scenarios remains a challenging task due to the following issues: (1) lack of public dataset annotation of group-housed pigs similar to MOT20 [32] datasets used for all MOT frameworks; (2) most of the detection algorithms are sensitive to variable lighting conditions and the occlusions of pigs in a commercial environment; (3) the error pig IDs in the tracking due to the variable farm environment and overlapping of one pig from another. To tackle these challenges, some works have proposed to develop the public dataset annotations of group-housed pigs. For example, the dataset annotations of group-housed pigs were presented [19]; however, it annotated the keys of pigs and was not suitable for mainstream pig behavior tracking algorithms. Due to the lack of public dataset annotation of growing pigs similar to MOT20, there are few studies on tracking pig behavior long-term (over an hour) and avoiding error IDs in pig tracking by using advanced tracking algorithms, such as DeepSORT, FairMot [33], and ByteTrack [34]. So, public dataset annotation of group-housed pigs is crucial in tracking pig behavior analyses tasks.

Based on these works, to void the error switches of pig ID during tracking using the basic DeepSORT algorithm, an improved DeepSORT algorithm was proposed. Firstly, the public dataset annotation of group-housed pigs similar to the MOT20 was developed using the original dataset [17]. Then, the detectors YOLOX-S and YOLO v5s were used to detect pigs and classify the four behaviors including standing, lying, eating, and other. Finally, the proposed approach was used to tackle error switches of pig ID in the tracking process by improving the trajectory processing and data association for the specific pig scenario. The merits of the improved DeepSORT approach contained two points. One was that we improved the trajectory processing and data association during tracking procedure for pig-specific scenarios where the number of pig target objects remained unchanged. This improvement reduced error switches of pig ID and enhanced the stability of the tracking. The other was that the behavior classification information from detectors was used in the tracking algorithm for completing behavior tracking. The comparative experiments demonstrated that the improved DeepSORT algorithm obtained stabler ID in pig behavior tracking than the original DeepSORT.

The main contributions of this paper are summarized as follows:

- (1) An improved DeepSORT algorithm was proposed to significantly decrease error switches of pig ID in the pig behavior tracking process.
- (2) The public dataset annotation of group-housed pigs similar to MOT20 was established for the development of advanced pig-tracking algorithms.
- (3) Two advanced detectors, YOLOX-S and YOLO v5, were used for pig target detection.
- (4) The tracking results of the improved DeepSORT based on YOLOX-S achieved a MOTA of 98.6%, and its IDs had a significant decrease, with a decline of 80% compared with DeepSORT.

## 2. Dataset Annotation of Group-Housed Pigs

# 2.1. DataSet Description

In this study, the original dataset was selected from the public pig videos [17], which contained a total of 15 videos of  $2688 \times 1520$  resolution with half an hour. The collection of videos depicted different animal ages/sizes, variations in housing facilities, basic activity levels, and lighting scenarios. Due to the camera's height and focal length, there were items captured from an overhead camera that did not belong to this pen, so, in the experiment, we used the video cropping method to remove items that were not in the pen and reduce the influence of the external environment. We chose 12 videos suitable for pig behavior tracking algorithms and classified the activity levels for the pigs as high (H), medium (M), and low (L) according to the classifications of the original dataset. These video sequences contained pigs lying, eating, drinking, standing, or engaging in other behaviors to meet the experimental requirements. Our dataset consisted of 12 videos, of which 8 videos were

used for training and 4 videos were used for testing. Table 1 shows the properties of the 12 videos annotated for pig behavior tracking and performance analysis.

	Early Finisher							Late Finisher					
Video#	Pig01	Pig02	Pig03	Pig04	Pig05	Pig06	Pig07	Pig08	Pig09	Pig10	Pig12	Pig15	
Day	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$		$\checkmark$	$\checkmark$		$\checkmark$		$\checkmark$		
Night					$\checkmark$			$\checkmark$		$\checkmark$		$\checkmark$	
# of Pigs	7	7	15	15	8	16	12	13	14	14	15	16	
Activity Level	Н	L	М	М	М	Н	М	L	М	М	L	Н	
Sparse/Dense	Spare	Spare	Dense	Dense	Spare	Dense	Dense	Dense	Dense	Dense	Dense	Dense	
Train/Test	Train	Test	Test	Train	Train	Train	Train	Train	Train	Test	Train	Test	

Table 1. Properties of 12 videos annotated for pig behavior tracking and performance analysis.

2.2. The Procedure of Dataset Annotation

The annotation process of the dataset in the experiment consisted of the following steps: (1) First, all video sequences were cropped by using FFmpeg software for tracking the pig behaviors in each pen. Part of the cropped dataset is shown in Figure 1. Figure 1a is a video sequence 02# daytime, sparse; Figure 1b is a video sequence 05# nighttime, sparse; Figure 1c is a video sequence 07# daytime, dense; and Figure 1d is a video sequence 10# nighttime, dense. After the videos had been cropped, there were no other irrelevant pigs or objects in each pen, and we could achieve a better tracking performance.



**Figure 1.** Part of the cropped dataset, (**a**) video sequence 02#, (**b**) video sequence 05#, (**c**) video sequence 07#, and (**d**) video sequence 10#.

(2) Then, DarkLabel software was used to label all video sequences; each video sequence of 1 min (5 frames per second) was 300 frames of images, and there were 3600 images in a total of 12 videos.

(3) Finally, using the annotation files produced by DarkLabel software and video sequences, we wrote Python script files to convert the annotation files into JASON files suitable for detection (COCO dataset) and reidentification (ReID) data suitable for DeepSORT tracking. Part of the dataset for pig detection and tracking is shown in Figure 2. Figure 2a,b show the behavior-annotated images of video sequences 09# and 10# for detection and behavior recognition. Figure 2c is part ReID dataset including 10 different pigs in day and night scenes. The feature maps of ReID dataset were used for appearance matching of pig behavior tracking.



**Figure 2.** Part dataset of detection and tracking. (**a**) behavior-annotated images of video sequence 09# for detection, (**b**) behavior-annotated images of video sequence 10# for detection, and (**c**) part ReID dataset including 10 different pigs in day and night scenes for DeepSORT pig tracking.

# 3. Methods

# 3.1. Process and Flow Chart

We used the pipeline in Figure 3 to train and validate our system. The pipeline was divided into two procedures. Firstly, the video dataset annotated using the four postures of lying, eating, standing, and other was fed to the developed CNN-based behavior detector for training. The detectors used included YOLOX-S and YOLO v5s due to their optimal speed and accuracy tradeoff for real-time applications. For both detectors, we used the same environment, e.g., data augmentation, neck network, and other hyperparameters. Then, the improved DeepSORT tracking algorithm was used to track detected pig objects between two successive frames. DeepSORT adopted Kalman filtering (KF), track management, and data association to preserve the identity of pigs across consecutive frame sequences. To prevent wrong switches of pig ID numbers due to complex scenarios, e.g., pig overlapping, occlusion, and illumination change, we improved the strategies of data association of DeepSORT to achieve stable pig ID numbers according to pig farm characteristics.

The YOLO (YOLOX-S or YOLO v5s) detectors and the improved DeepSORT algorithms together accomplished individual pig behavior recognition and tracking under natural pig scenarios. In the next section, we described the training process of the detector and pig-tracking algorithm.

# 3.2. YOLO v5 and YOLOX-S Target Detection Network

YOLO v5 is the advanced detection model of the YOLO series that uses a single multilayer network to predict the bounding box and classification probability. In the COCO dataset, the YOLO v5 algorithm has shown remarkable accuracy and high detection speed; therefore, the YOLO v5 network was used for group-housed pig detection. According to different network depths and widths, YOLOv5 can be divided into four basic network structures: YOLO v5s, YOLOv5m, YOLOv5l, and YOLOv5x. The YOLO v5s model with the least number of parameters was chosen for target detection.



**Behaviours Tracking Results** 

Multi-Object Tracking

Figure 3. Architecture proposed for automated behavior recognition and tracking of group-house pigs.

YOLOX is a high-performance detector proposed in 2021 that combines excellent advances in target detection, such as decoupled head, strong data enhancement, anchorfree, and simOTA with YOLO. YOLOX not only surpasses the AP of YOLOv3, YOLov4, and YOLOv5, but also achieves extremely competitive inference speeds. According to its scaling rule, YOLOX mainly includes YOLOX-S, YOLOX-M, YOLOX-L, and YOLOX-X models. We also chose YOLOX-S with the least number of parameters for comparison with the YOLO v5s model.

The structures of YOLO v5s and YOLOX-S are mainly divided into four stages [25]. The first stage of the model is called input procedure, which includes scaling transformation, color space adjustment, and mosaic data enhancement to increase the amount of data for training to improve the robustness of the model. The second stage is called backbone network, which uses the CSPDarkNet53 as a basic network for feature extraction. The third stage is called detection neck (PAFPN) that adopts the feature pyramid network (FPN) and the path aggregation network (PAN) structure to retain the rich spatial information from the bottom-up data and semantic information from the top-down data steam. The fourth stage is called decoupled head (Prediction), which is used for object detection and classification to improve the converging speed and accuracy.

# 3.3. DeepSORT for the Group-Housed Pig Tracking Model

DeepSORT is an improved version of the SORT algorithm, which includes data association, KF estimation, and track management for multi-object tracking. It enhances the matching effect of targets by combing appearance and motion information to reduce the number of ID switches under an occlusion environment. Moreover, DeepSORT is the method of TBD, which first uses the detector to detect and classify objects, then performs the data association over consecutive frames, and then outputs the classification, location, and track ID information. The DeepSORT tracking algorithm is represented in Algorithm 1.

Algorithm 1: DeepSORT algorithm
: Define DeepSORT configurations as cfg 2: Initialize DeepSORT (DeepSORT, cfg)
B: Initialize device(gpu)
l: Model = load(detection model, device = gpu)
: Dataset = loadimgaes(input videos)
5: For i, img in enumerate(dataset):
7: pred = model(img)
B: pred = non_max_suppression(pred)
error i, detections enumerate(pred):
0: xyxy, conf, cls = detections
1: Pass detection results to DeepSORT
2: features = get_features(xyxy, img)
3: Predict track by using Kalman filter
4: Compute cost matrix using features, track, detections
5: matches, unmatched_tracks, unmatched_detections = match(cost matrix)
6: Update track set by using match results
7: Save results

In the DeepSORT algorithm, there are two state vectors named detection and track. Detections are used to store the detected bounding boxes (BB)  $(D_i, i \in \{1...N\})$  from the current frame by the object detector, and tracks are regarded as the correctly matched tracks  $(T_i, i \in \{1...M\})$ , which include the position, state, and speed information of the targets before the current frame, respectively. Among them,

$$D_i = \{(top, left, width, height), conf, feature, classes\}$$
(1)

where *top*, *left*, *width*, and *height* denote coordinates of the upper left point, the width and height of the bounding box. *conf* denotes the confidence; *feature* and *classes* denote reidentification features of the detected object used for cascade matching and classification information.

$$T_i = \{m, c, Track\_ID, hits, time, state, features, classes\}$$
(2)

where *m*, *c* denote tracks' mean (including location) and variance for KF prediction and update; *Track\_ID*, *hits*, *time*, and *state* denote tracking ID number, number of matches, plus 1 if successful, number of recent KF updates, tracks' state including tentative, confirmed, and deleted. *features* and *classes* are similar to the corresponding detection.

The key three modules of DeepSORT in Figure 3 are described: (1) The data association module is responsible for matching the KF's predicted bounding boxes with detections on the image. The association of detections to tracks is solved by the Hungarian algorithm, using cascade and IOU matching. Firstly, the DeepSORT method uses cascade matching including motion and appearance metrics to associate valid tracks. The second part uses intersection over union (IoU) to associate unmatched and tentative tracks (recently created) with unmatched detections.

(2) The KF estimation module uses a linear constant velocity model to represent each track motion model. When a detection is associated with a tracked object (track), its BB is used to update the track state. If no detection is associated with the track, then the track's state is only predicted.

(3) The track management module is used for the creation and deletion of tracks. New tracks are created when detections do not overlap or overlap with tracks below a minimum IoU threshold. The BB of the detection is used to initialize the KF state. If a new track does not receive updates because it does not receive associations or if tracks stop receiving associations, they are deleted to avoid maintaining a high number of tracks to false positives or objects that left the scene.

#### 3.4. Improved DeepSORT Method

In the group-housed pig tracking application, as the video frames grow, DeepSORT will assign different ID to the same pig target, and the maximum ID value of the pig will

significantly exceed the number of real pig targets. In addition, the pig ID are changed wrongly in tracking; the main reason is that the detection results cannot match to the original tracks when the target pigs are moving or overlapping due to occlusion, resulting in new tracks generated from unmatched detection results. To overcome these problems, we proposed an improved DeepSORT, which was used to limit the target object ID growth for pig-specific scenarios and improve matching process and track generation in the tracking process. Figure 4 shows the flow chart of tracking process of the improved DeepSORT.



Figure 4. Flow chart of the tracking process of the improved DeepSORT.

The improved DeepSORT includes two parts:

(1) Limiting the target object ID growth for pig-specific scenarios.

In Figure 4, the improvement 1 of limiting the target object ID growth contains two subpoints. Firstly, the number of detections (n) in each pen obtained through the detector was stored in a one-dimensional array with a length equal to 3, which was called the target number array (TNA). If the video frame was the first frame, all the values of the array were set to the number of detected targets (n1 = n2, n2 = n3) in the first frame. In consecutive frame sequences, the number of detection targets in the current frame sequence was inserted into the end of the array, and the number of targets stored in the head of the array was deleted. The process of initialization and updating the target number array is shown in Figure 5.



Figure 5. Initialization and updating of the target number array.

Secondly, because the detection result had a certain probability of false and missed detection, we created the tracks for the detections after the second round IOU match failure according to the TNA. The process of a new track creation contained three steps:

(a) For frame T - 1, the TNA dynamically stored the total number of targets in the detection result across consecutive frame sequence (Length = 3, in this study), and the ID maximum value was determined by the rounded value of the average of the TNT. Suppose the TNT was set to [15,15,15], then the ID maximum value was 15.

(b) For frame T, the detector detected a new target, and the ID of the unmatched detection box was 16; the TNT was set to [15,15,16] according to the rule of Figure 5, and then the ID maximum value was 15. If the current generated unmatched detection ID

value (16) exceeded the ID maximum value (15), it was considered a false positive and no new track is created.

(c) For frame T + 1, the detector detected a new target in two consecutive frames; the ID of the unmatched detection box was 16; the TNT was set to [15,16,16], and then the ID maximum value was 16. If the current generated unmatched detection ID value (16) did not exceed the maximum ID value (16), it was considered a false negative and a new track was created.

(2) Adding the second round of IOU matching to associate the unmatched detections and tracks.

Because the pigsty was a specific and closed scene, usually no pig target was added or reduced. To improve the matching effect during the data association process, after the first round of IOU matching was finished for initial matching, we added the second round of IOU matching to deal with the unmatched detections. Figure 4 shows the improvement 2 for the process of the second round of IOU matching. Meanwhile, the second round of IOU matching set the IOU to a larger value, and it was possible to achieve better results in associating unmatched detections with tracks.

Finally, the tracking process of the improved DeepSORT in Figure 4 is listed as follows:

(1) The target number array was dynamically created according to the results of the detection of video sequences.

(2) Cascade matching was used for the first matching of detections and confirmed tracks during consecutive frame sequence, and the matching tracks were updated with KF based on assigned detections.

(3) Unconfirmed tracks (before frame), unassigned tracks in cascade matching, and unmatched detections were matched for the first round of IOU matching using the Hungarian assignment algorithm.

(4) The unmatched tracks and unmatched detections produced after the first round of IOU matching were used for the second round of IOU matching operations. For unmatched detections, we used the rule of the improvement 1 to create the new tracks. For unassigned tracks, we removed the tracks that met the conditions to be deleted.

#### 3.5. The Evaluation Metrics

The following evaluation criteria were used to evaluate the results of multiobject tracking model.

$$Precision = \frac{TP}{TP + FP} Recall = \frac{TP}{TP + FN} F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$
(3)

where True Positive (TP) is the number of pixels correctly predicted to the pig category; False Positive (FP) is the number of pixels incorrectly predicted to the pig category; False Negative (FN) is the number of pixels predicted to the pig category as the background, and F1 is a comprehensive evaluation index of precision and recall rate.

Multi-object tracking accuracy (MOTA) measures the performance of tracking in detecting objects and maintaining trajectories, and it was calculated as follows:

$$MOTA = 1 - \frac{\sum (FN + FP + IDs)}{\sum GT} \in (-\infty, 1]$$
(4)

where *IDs* is the ID switch, and *GT* is the number of all objects.

Multiple object tracking precision (MOTP) indicates the positioning accuracy of the detector, and it was calculated as follows:

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{c_t}$$
(5)

where *d* is the average metric distance, and *c* denotes the number of successful matches for the current frame.

ID switch (IDs) refers to the total number of ID switches in the video. The lower the value, the better the performance.

Mostly tracked (MT) denotes the number of successful tracking results that matched the true value at least 80% of the time.

Identification precision (IDP), Identification recall (IDP), and Identification F1(IDF1) denote ID precision, ID recall, and IDF1 score similar to the precision, recall, and F1 score of the metrics in the detection model, and the main role of these metrics was to evaluate the performance of ID switches. The higher their values, the better their performance.

# 4. Experimental Section

# 4.1. Experiments Configuration

In this study, the experiments of five types were conducted for behavioral track-ing of group-housed pigs: (1) Object detection experiments using YOLOX-S or YOLO v5s detectors; (2) The pig behavior tracking experiments using the improved and orig-inal DeepSORT algorithm based on YOLOX-S; (3) The comparative tracking experiments of improved and original DeepSORT model based onYOLO v5s; (4) The com-parative tracking results for day and night conditions between the improved and original DeepSORT based on YOLOX-S and YOLO v5s models; (5) The compara-tive experiments between the improved DeepSORT+YOLOX-S model and other two advanced MOT methods. The YOLOX-S and YOLO v5s were running combined with improved DeepSORT using the same configuration parameters.

For the detection model, the origin image's resolution was  $2688 \times 1520$  pixels, and after the preprocessing process of the input part of the detection model, its resolution was uniformly adjusted to  $640 \times 640$  pixels to achieve faster processing speed. A total of 4620 images of group-housed pigs were used totally for training. The number of output categories of the network was set to 4 (lying, eating, standing, and other). The initial learning rate was set to 0.01, while the IoU, epoch, confidence, training batch, and batch size were set to 0.45, 200, 0.4, 64, and 16, respectively. The optimizer was SGD.

For the ReID model, the dataset contained 137 different pigs with an average of 300 images per pig, randomly divided into training and testing sets according to a 7:3 ratio.

For the tracking model, the annotated images from 12 videos were used for training and testing according to Section 2.2. Among them, 8 videos were used for training of the DeepSORT model. The remaining four video sequences were used for the test, and they were called Pig02, Pig03, Pig10, and Pig15, where the number of pigs in Pig02 was less than 10 and Pig02 was obtained at day. The number of pigs in Pig03, Pig10, and Pig15 was more than 10, where Pig10 and Pig15 were obtained at night and Pig03 was obtained at day. Table 1 shows the detailed information on the training and test dataset in the tracking model.

#### 4.2. Results and Analysis of Pig Detection and Behavior Classification

To judge the model training process and the effectiveness of object detection in these experiments, precision and recall, P–R curve, mAP@0.5, and mAP@0.5:0.95 [35] of the test dataset were used as the main parameters to judge the performance of detection model.

Figure 6 shows the P–R curves of the four classes of lying, standing, eating, and other and their average precision values (mAP) at IoU with 0.5 using five different colored curves for the YOLO v5s (Figure 6a) and YOLOX-S (Figure 6b) models. The area of the P–R curve of YOLO v5s was larger than that of YOLOX-S, which indicated that the YOLO v5s model had better performance than the YOLOX-S model. Meanwhile, both models had a high precision and recall for all four categories and a low false detection rate, so the two models could be used as the detection models for all animal behaviors.



**Figure 6.** P–R curves of YOLO v5s and YOLOX-S models. (**a**) The P–R curve of YOLO v5s. (**b**) The P–R curve of YOLOX-S.

The experimental results of the different methods on object detection are shown in Table 2. The number of test images was 660 in Table 2, and the total number of labeled pigs was 8047, of which the percentage of lying, standing, eating, and other were 56.77% (4568/8047), 22.52% (1812/8047), 15.63% (1258/8047), and 5.08% (409/8047), respectively. YOLO v5s and YOLOX-S both achieved a precision rate of over 97% and a recall rate of over 98% in all categories, with a mAP (IoU = 0.5) rate of over 98.4%, while the IoU was set at 0.5, and there was an average mAP@0.5:0.95 rate of over 87%, while the mAP@0.5:0.95 rate was the average value of the APs with the IoU thresholds from 0.5 to 0.95 with an interval of 0.05. The average inference time of YOLOX-S and YOLO v5s was 0.012s and 0.017s, respectively. According to experimental results, we found that YOLO v5s achieved a better performance in precision, mAP@0.5, and mAP@0.5:0.95 compared to YOLOX-S.

Method	Class	Labels	Precision (%)	Recall (%)	F1 (%)	mAP@0.5	mAP@0.5:0.95
	lying	4568	100	98.1	99.0	98.9	94.9
	standing	1812	99.6	99.2	99.4	99.7	87.4
YOLO v5s	eating	1258	99.7	99.5	99.6	99.8	86.9
	other	409	98.2	99.3	98.7	99.5	88.1
	all	8047	99.4	99.0	99.2	99.5	89.3
	lying	4568	98.0	98.6	98.3	98.0	94.0
	standing	1812	99.0	99.7	99.3	99.0	85.0
YOLOX-S	eating	1258	98.8	99.8	99.3	98.8	83.1
	other	409	97.9	98.8	98.3	97.9	85.8
	all	8047	98.4	99.2	98.8	98.4	87.0

Table 2. Experimental results of YOLOX-S and YOLO v5s models.

Figure 7 shows the detection performance of different algorithms on the same data. Figure 2a,b show the GT (Ground Truth) information for all behavior classes of videos 09# and 10#. The left and right columns of Figure 7 are the detected results under day and night illustrations. Both models detected all pigs under complex scenes compared with the information of the GT.



Figure 7. Detection performance of different detection models. (a) YOLO v5s and (b) YOLO X-S.

# 4.3. The Tracking Results of Improved and Original DeepSORT Based on YOLOX-S

Based on the eight metrics of MOT (Section 3.5), the tracking results of the improved and original DeepSORT based on YOLOX-S are listed in Table 3. In Table 3, the GT denoted the number of pigs in each pen, and the GT and MT in all test videos were equal meaning that there was no lost target during video tracking. The IDF1, IDP, IDR, and MOTA values of the original DeepSORT were 88.9%, 88.5%,96.8%, and 95.1%, respectively. Meanwhile, the improved DeepSORT based on YOLOX-S obtained MOTA of 98.6%, MOTP of 95.5%, IDP of 96.2%, IDR of 95.3%, and IDF1 of 95.7%. Compared with the original DeepSORT (Table 3, the last row), the values of MOTA, MOTP, IDP, IDR, and IDF1 of the improved DeepSORT increased by 1.8%, 0.4%, 7.7%, 7%, and 6.8%, respectively. In addition, the FP and IDs values of the improved algorithms were only 21 and 15, while they reached 266 and 75 using the original DeepSORT algorithm, which further demonstrated that the improved DeepSORT algorithm effectively reduced the frequent error switches of pig ID and improved the accuracy of the tracker.

Method	Name	GT	MT↑	FP↓	IDs↓	IDF1↑	IDP↑	IDR↑	MOTA↑	MOTP↑
YOLOX-S + Improved DeepSORT	Pig02	7	7	0	0	99.7%	100%	99.3%	99.3%	97.3%
	Pig03	15	15	21	13	88.7%	89.5%	88.0%	97.1%	93.8%
	Pig10	14	14	0	2	96.7%	97.1%	96.3%	99.1%	96.1%
	Pig15	16	16	0	0	99.7%	100%	99.3%	99.3%	95.9%
	Ăll	52	52	21	15	95.7%	96.2%	95.3%	98.6%	95.5%
	Pig02	7	7	2	1	99.1%	99.4%	98.8%	99.2%	97.3%
VOLOV &   Original	Pig03	15	15	222	38	80.0%	78.7%	81.3%	92.6%	92.5%
DeepSORT	Pig10	14	14	27	27	90.6%	90.6%	90.5%	98.0%	96.0%
	Pig15	16	16	15	9	91.4%	91.5%	91.2%	98.8%	95.9%
	Ăll	52	52	266	75	88.9%	88.5%	89.2%	96.8%	95.1%

Table 3. Tracking results of improved and original DeepSORT based on YOLOX-S.

Figures 8 and 9 show the tracking comparison results for Pig03 and Pig15 using the original and improved DeepSORT based on YOLOX-S. In Figure 8c,d, the maximum ID value remained at 17, while it reached 25 and 34 using the original DeepSORT in Figure 8a,b. The ID value of 15 (shown in Figure 8c,d) was stabilized at 15 from frame 50 to frame 150, and no further changes were produced using the improved DeepSORT. The experimental results demonstrate that the improved DeepSORT outperformed DeepSORT in the ID switches under heavy pig overlapping conditions.



**Figure 8.** Tracking results of original and improved DeepSORT based on YOLOX-S model of Pig03. (a) Frame 50, original DeepSORT; (b) Frame 150, original DeepSORT; (c) Frame 50, improved DeepSORT; and (d) Frame 150, improved DeepSORT.



**Figure 9.** Tracking results of original and improved DeepSORT based on YOLOX-S model of Pig15. (a) Frame 50, original DeepSORT; (b) Frame 150, original DeepSORT; (c) Frame 50, improved DeepSORT; and (d) Frame 150, improved DeepSORT.

From Figure 9a,c, in frame 50, the improved DeepSORT and DeepSORT achieved the same tracking results with the same maximum ID value of 16. However, in frame 150, the maximum ID value of DeepSORT reached 17 (as shown in Figure 9b), while that of the improved DeepSORT remained at 16 in Figure 9d.

# 4.4. The Tracking Results of Improved and Original DeepSORT Based on YOLOXv5s

This improved algorithm achieved significant tracking performance gains not only using the YOLOX-S detector, but also using the YOLO v5s detector. Table 4 shows the tracking results of the improved DeepSORT algorithm based on the YOLO v5s model. The IDF1, IDP, IDRm, and MOTA values of the improved algorithm were 94.6%, 96.1%, 93.2%, and 96%, respectively, while using the original DeepSORT algorithm, the values of IDF1, IDP, IDR, and MOTA were 80.8%, 81.0%, 80.5%, and 94.6%, respectively. Moreover, the FP and IDs values of the improved algorithms were only 73 and 12, while they reached 319 and 111 using the original DeepSORT, which further demonstrated that the improved DeepSORT algorithm effectively improved the frequent error switches of ID and the accuracy of the tracker.

Method	Name	GT	MT↑	FP↓	IDs↓	IDF1↑	IDP↑	IDR↑	MOTA↑	MOTP↑
YOLO v5s + Improved DeepSORT	Pig02	7	7	0	0	99.7%	100%	99.3%	99.3%	97.3%
	Pig03	15	13	71	7	84.6%	87.8%	81.6%	89.6%	90.0%
	Pig10	14	14	2	5	96.9%	97.9%	95.9%	97.7%	95.9%
	Pig15	16	16	0	0	99.6%	100%	99.1%	99.1%	96.6%
	All	52	50	73	12	94.6%	96.1%	93.2%	96.0%	94.7%
YOLO v5s + Original DeepSORT	Pig02	7	7	6	4	91.8%	92.0%	91.6%	98.9%	97.3%
	Pig03	15	14	178	38	79.1%	80.4%	77.7%	87.9%	91.9%
	Pig10	14	14	49	40	80.1%	79.9%	80.2%	97.0%	96.2%
	Pig15	16	16	86	29	78.2%	77.8%	78.6%	96.9%	96.7%
	Âll	52	51	319	111	80.8%	81.0%	80.5%	94.6%	95.3%

Table 4. Tracking results of improved and original DeepSORT based on YOLO v5s.

Figure 10 shows the tracking comparison results for video Pig03 using the improved and original DeepSORT based on the YOLO v5s detector. The number of pigs in the video Pig03 was 15. At frames 50, 100, and 150, DeepSORT (Figure 10a, left column) achieved the maximum ID values of 19, 25, and 33, respectively, while the improved DeepSORT (Figure 10b, right column) obtained the maximum ID values of 14, 15, and 15. So, we found that the improved DeepSORT avoided the error switches of ID values and maintained stable ID values compared to DeepSORT.



**Figure 10.** Results comparison between DeepSORT and improved DeepSORT based on YOLO v5s model of Pig03. (a) DeepSORT, at frames 50, 100, and 150; and (b) Improved DeepSORT, at frames 50, 100, and 150.

# 4.5. The Comparative Tracking Results for Day and Night Conditions

Figures 11 and 12 show the tracking results for the day and night conditions using the improved and original DeepSORT based on YOLOX-S and YOLO v5s models, respectively. For the day, Pig02 was selected for comparison. In Figure 11a–c, the maximum ID remained 7 at frames 50, 100, and 150 of Pig02, and the ID value did not switch as in Figure 11b,c when the pigs walked.



**Figure 11.** Tracking results for Pig02 using improved and original DeepSORT based on YOLOX-S model for day condition. (**a**–**c**) show the results of improved DeepSORT, and (**d**–**f**) show the results of original DeepSORT. (**a**) Improved model, frame 50; (**b**) Improved model, frame 100; (**c**) Improved model, frame 150; (**d**) original, frame 50; (**e**) original, frame 100; and (**f**) original, frame 150.



**Figure 12.** Tracking results of Pig 10 using improved and original DeepSORT based on YOLO v5s model for night condition. (**a**–**c**) show the results of improved DeepSORT, and (**d**–**f**) show the results of original DeepSORT. (**a**) Improved model, frame 50; (**b**) Improved model, frame 100; (**c**) Improved model, frame 150; (**d**) Frame 50, Pig10; (**e**) Frame 100, Pig10; and (**f**) Frame 150, Pig10.

For night, Pig 10 was selected for comparison. In Figure 12a–c, ID8 (standing) of Pig10 still kept its ID value during walking, unlike in Figure 12f where many ID exchanges occurred when the pigs were crowded together at frame 150, and ID1 (lying) in Pig 10 did not change from frame 50 to frame 150, while the same pig (ID1) in Figure 12d grew to ID18 (lying) at frame 100 to frame 150 as shown in Figure 12e, f. In the top left corner of Figure 12, the unshown behavior and IDs of the three pigs of Pig 10 were lie2, lie3, and lie5. The tracking results show that the improved DeepSORT model achieved better performance than the original DeepSORT model.

# 4.6. Results Comparison with Other Advanced MOT Methods

As the YOLOX-S + Improved DeepSORT model (Table 3) achieved better performance than the YOLO v5s + Improved DeepSORT model (Table 4) during pig tracking, we used the improved YOLOX-S + Improved DeepSORT model to conduct comparison experiments with the two advanced MOT methods including JDE and TrasnTrack on the same dataset. The JDE [27] proposed by Wang et al. is an MOT system that allows target detection and appearance embedding to be learned in a single-shot deep network, which is the first (near) real-time MOT system, with a running speed of 22 to 40 FPS depending on the input resolution. TrasnTrack [28] builds up a novel joint detection-and-tracking paradigm by accomplishing object detection and object association in a single shot, which leverages the transformer architecture.

16 of 20

Table 5 shows the experimental results of the comparison by using our approach and the other two methods. Our approach achieved the best performance on the eight metrics of MOT (Section 3.5). The IDF1, IDP, IDR, IDs, and MOTA values of our approach were 95.7%, 96.2%, 95.3%, 15, and 98.6%, respectively. Compared with the JDE method, our approach improved by 10.6% and 18% in MOTA and IDF1, respectively, and decreased by 150 in the IDs value. Compare with TransTrack, the values on the MOTA and IDF1 metrics of our approach were also significantly improved by 7.1% and 14.3%, respectively. In addition, the FP and IDs values of TransTrack were 439 and 142 higher than that of our approach. These comparison results further demonstrate that our approach can effectively reduce the frequent error switches of pig ID and improved the tracking performance.

Method	Name	GT	MT↑	FP↓	IDs↓	IDF1↑	IDP↑	<b>IDR</b> ↑	<b>MOTA</b> ↑	MOTP↑
	Pig02	7	6	21	14	77.0%	78.2%	75.9%	97.4%	99.2%
	Pig03	15	14	41	10	90.4%	93.9%	87.1%	90.7%	80.1%
JDE	Pig10	14	10	79	67	70.5%	75.4%	66.2%	82.5%	87.1%
	Pig15	16	14	136	74	72.2%	74.3%	70.2%	87.3%	84.6%
	Ăll	52	44	277	165	77.7%	80.8%	74.8%	88.0%	88.4%
	Pig02	7	7	12	2	91.9%	92.5%	91.3%	97.4%	91.5%
	Pig03	15	14	55	16	94.4%	94.9%	94.0%	96.3%	93.1%
TransTrack	Pig10	14	13	135	58	67.3%	68.1%	68.1%	89.9%	89.2%
	Pig15	16	14	237	66	76.7%	77.8%	76.7%	85.9%	82.4%
	All	52	48	439	142	81.4%	82.2%	80.6%	91.5%	88.6%
	Pig02	7	7	0	0	99.7%	100%	99.3%	99.3%	97.3%
VOLOV 6   Improved	Pig03	15	15	21	13	88.7%	89.5%	88.0%	97.1%	93.8%
DoorsOPT (our op prooch)	Pig10	14	14	0	2	96.7%	97.1%	96.3%	99.1%	96.1%
DeepSOK1 (our ap-proach)	Pig15	16	16	0	0	99.7%	100%	99.3%	99.3%	95.9%
	Ăll	52	52	21	15	95.7%	96.2%	95.3%	98.6%	95.5%
TransTrack YOLOX-S + Improved DeepSORT (our ap-proach)	Pig03 Pig10 Pig15 All Pig02 Pig03 Pig10 Pig15 All	$ \begin{array}{c} 15\\ 14\\ 16\\ 52\\ \hline \\ 7\\ 15\\ 14\\ 16\\ 52\\ \end{array} $	14 13 14 48 7 15 14 16 52	55 135 237 439 0 21 0 0 21 21	16 58 66 142 0 13 2 0 15	94.4% 67.3% 76.7% 81.4% 99.7% 88.7% 96.7% 99.7% 95.7%	94.9% 68.1% 77.8% 82.2% 100% 89.5% 97.1% 100% 96.2%	94.0% 68.1% 76.7% 80.6% 99.3% 88.0% 96.3% 99.3% 95.3%	96.3% 89.9% 85.9% 91.5% 99.3% 97.1% 99.1% 99.3% 99.3% 98.6%	93.1% 89.2% 82.4% 88.6% 97.3% 93.8% 96.1% 95.9% 95.5%

Table 5. Experimental results of the comparison with the other two methods.

#### 5. Discussion

In real farming scenarios, the identity of pigs is hard to track due to dense overlapping and occlusion, which still make it challenging to automatically track the behavior of group-housed pigs using computer vision techniques. To improve the problem, this study proposed an improved DeepSORT algorithm of behavior tracking based on the YOLOX-S and YOLO v5s detectors.

The improved DeepSORT contained two innovation points. An innovative point was limiting the target object ID growth. For the specific group-housed pig farms, where the number of pigs in each pen did not change, we proposed improving the trajectory processing to create tracks. Pig targets in each pen were detected, and the total number of targets in the detection result was stored dynamically, and then the numbers of detected targets in the last three consecutive frames were saved as the target number array and used for improving the trajectory management. The created track ID numbers did not exceed the maximum number. Another innovative point was adding the second round of IOU matching to associate the unmatched detections and tracks. Thus, the improved DeepSORT algorithm avoided the error switches of every pig ID in each pen and improved the accuracy of the tracker. In the tracking process, the pig behavior category was added to the track of target tracking, which accomplished the tracking of the pig behavior category. In addition, it could automatically count the different behavior times of each pig every day in each pen and was used to identify whether the pig was abnormal in health.

In terms of behavior tracking, Alameer et al. (2020) adopted the Faster R-CNN and YOLOv2 as the detectors and DeepSORT as the tracker to overcome illumination changes and the occlusions of pigs in the commercial environment. In addition, a deep-learning-based pig posture and locomotion activity detection and DeepSORT tracking algorithms were designed to measure pig behavior changes in an experimental pig barn at different

greenhouse gas (GHG) levels [18]. These approaches were good at detection and tracking behaviors. However, as the video frames grew and there were error pigs IDs in tracking due to dense overlapping and occlusion, so the DeepSORT algorithm could hardly be used to track different pig behaviors over a long period. Our proposed improved DeepSORT could avoid the error changes of ID, which is suitable for monitoring the different behaviors of pigs in long-time tracking.

Moreover, the dataset [16] for tracking was specifically annotated. Instead of annotating pig postures in each frame, unique identification numbers were given to each pig across the frames to avoid error ID switches. The annotated dataset was not suitable for target detection and tracking by the public tracking frameworks. However, our dataset annotation of group-housed pigs was established similarly to MOT20 using DarkLabel software, which can be used for the development of many advanced tracking algorithms for pig behavior tracking.

Our proposed method still has some limitations. Compared with JDE methods [27], the improved DeepSORT is still a TBD paradigm, which runs slowly and cannot achieve real-time performance; Furthermore, in long-time tracking, detections with a low score in the detector are removed, which will lead to the erroneous deletion of some tracks. The reason is that low confidence detection boxes sometimes indicate the existence of objects, e.g., the occluded objects. Filtering out these objects causes irreversible errors for MOT and brings non-negligible missing detection and fragmented trajectories.

With significant improvements in the performance of target detectors based on deep learning, YOLOX-S, or YOLO v5s, the combined tracker achieved advanced detection and tracking performance for pig behavior tracking in complex scenes. The improved tracking algorithm significantly reduced IDs errors and achieved stable individual pig tracking according to real farming scenarios. Thus, the proposed algorithm is an effective solution for automatically detecting and tracking multiple pigs, and it will qualify the current application for tracking pig behavior quickly and accurately.

# 6. Conclusions and Future Directions

This paper proposed an improved DeepSORT algorithm for automated behavior recognition and tracking of group-housed pigs based on the YOLOX-S or YOLO v5s detectors. The DeepSORT was improved to reduce the error pig IDs significantly in tracking due to dense overlapping and occlusion and enhance the quality of tracking for pig-specific scenarios. In the detection experiments, the YOLO v5s network model was superior to the YOLOX-S network, Compared with YOLOX-S, the detection mAP@0.50:0.95 of YOLO v5s increased by 2.3%. However, in the tracking experiments, the YOLOX-S + DeepSORT model outperformed the YOLO v5s + DeepSORT model. Compared with YOLO v5s + DeepSORT, its MOTA, IDP, IDR, and IDF1 increased by 2.2%, 7.5%, 8.7%, and 8.1%, respectively, and the average detection and tracking time decreased by 0.005 s. Moreover, using the improved DeepSORT based on the YOLOX-S or YOLO v5s detectors, we found that the values of IDF1, IDP, and IDR achieved a very significant improvement, and the error switches of pig ID were effectively avoided for obtaining stable tracking of individual pigs. So, the improved behavior tracking algorithm can meet the needs in the actual farming environment and provide technical support for the contactless and automated monitoring of pigs, which has good engineering application prospects for the development of smart pig management. Further work consists mainly of the following two points:

(1) The daily behavior of each pig per pen can be automatically obtained by using an improved tracking algorithm, which can be used to analyze and find abnormal pig behavior. Therefore, we will conduct pig behavior analyses that assist the manual observations and improve pig farm automation management.

(2) Long-time tracking requires more resources and time, which is not sufficient for practical application. In the future, to achieve better speeds, we will utilize smaller detectors and the JDE trackers to complete pig behavior tracking. Moreover, to reduce storage capacities, we will probably use lightweight real-time object detectors, such as YOLOX-Tiny and fast-tracking methods to implement pig behavior tracking.

(3) MOT technologies have a wide range of applications, including intelligent video surveillance, military, automated driving, virtual reality, medical, health, and other fields. For example, in the auto drive system, the MOT algorithm tracks the movement of moving vehicles and pedestrians and predicts their future position, speed, and other information to accomplish automated driving. In the field of virtual reality, MOT technology uses the information, including human actions and continuous tracks captured by the camera, to achieve human–computer interaction. In the medical and health fields, with the help of medical auxiliary technologies, such as nuclear magnetic resonance and drug-targeting technology, the patient's lesion location is tracked to observe whether the lesion location has spread by using advanced MOT technology.

Further improvements will be considered by exploring a more effective detection model, such as YOLOv7 for detecting objects with fewer misses and less error detection due to influences related to occlusion, small target, and illumination changes. The strategies to mitigate the detection errors of detectors can improve the tracking performance. At present, to make the MOT method run in real time, a lot of work on designing the lightweight network structure of MOT system has been performed for embedded applications. Furthermore, designing and using efficient data association algorithm are also an effective strategy to improve target tracking in single cameras and multiple cameras. For example, the approaches of multilevel dynamic matching are designed for similarity matching and data association, which can effectively reduce the impact of the unstable factors of the detector on the tracker.

Author Contributions: Conceptualization, S.T. and X.L.; Data curation, S.W.; Funding acquisition, Y.L.; Investigation, L.H.; Methodology, S.T.; Project administration, Q.Z., Y.L. and Q.H.; Resources, Q.Z. and Y.L.; Software, X.L.; Visualization, Q.Z. and L.H.; Writing—original draft, S.T.; Writing—review & editing, S.T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China (61772209 and 31600591), the Science and Technology Planning Project of Guangdong Province (Grant No. 2019A050510034), College Students' Innovation and Entrepreneurship Competition (202110564025), Guangzhou Key Laboratory of Intelligent Agriculture (201902010081), and the key R&D project of Guangzhou (202206010091).

**Institutional Review Board Statement:** The animal study protocol was approved by the Institutional Review Board (or Ethics Committee) of the Guangdong Provincial Laboratory Animal Welfare and Ethical Review Guidelines and were approved by the Animal Welfare Committee of South China Agricultural University (No: 2021F129).

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author. The data are not publicly available due to restrictions eg privacy or ethical.

Acknowledgments: We would like to thank Yueju Xue for kindly providing the advice for this research.

Conflicts of Interest: The authors declare no conflict of interest.

#### References

- 1. Rowe, E.; Dawkins, M.S.; Gebhardt-Henrich, S.G. A Systematic Review of Precision Livestock Farming in the Poultry Sector: Is Technology Focussed on Improving Bird Welfare? *Animals* **2019**, *9*, 614. [CrossRef]
- 2. Chen, C.; Zhu, W.; Norton, T. Behaviour recognition of pigs and cattle: Journey from computer vision to deep learning. *Comput. Electron. Agric.* **2021**, *187*, 106255. [CrossRef]
- 3. Camerlink, I.; Scheck, K.; Cadman, T.; Rault, J.-L. Lying in spatial proximity and active social behaviours capture different information when analysed at group level in indoor-housed pigs. *Appl. Anim. Behav. Sci.* **2022**, 246, 105540. [CrossRef]
- Pandey, S.; Kalwa, U.; Kong, T.; Guo, B.; Gauger, P.C.; Peters, D.J.; Yoon, K.J. Behavioral Monitoring Tool for Pig Farmers: Ear Tag Sensors, Machine Intelligence, and Technology Adoption Roadmap. *Animals* 2021, *11*, 2665. [CrossRef]
- Zheng, W.; Tian, X.; Yang, B.; Liu, S.; Ding, Y.; Tian, J.; Yin, L. A Few Shot Classification Methods Based on Multiscale Relational Networks. *Appl. Sci.* 2022, 12, 4059. [CrossRef]

- Zheng, W.; Liu, X.; Yin, L. Research on image classification method based on improved multi-scale relational network. *PeerJ Comput. Sci.* 2021, 7, e613. [CrossRef]
- Zheng, W.; Yin, L.; Chen, X.; Ma, Z.; Liu, S.; Yang, B. Knowledge base graph embedding module design for Visual question answering model. *Pattern Recognit.* 2021, 120, 108153. [CrossRef]
- Ma, Z.; Zheng, W.; Chen, X.; Yin, L. Joint embedding VQA model based on dynamic word vector. *PeerJ Comput. Sci.* 2021, 7, e353. [CrossRef]
- 9. Zheng, W.; Yin, L. Characterization inference based on joint-optimization of multi-layer semantics and deep fusion matching network. *PeerJ Comput. Sci.* 2022, *8*, e908. [CrossRef]
- Zhang, H.; Luo, G.; Li, J.; Wang, F.-Y. C2FDA: Coarse-to-Fine Domain Adaptation for Traffic Object Detection. *IEEE Trans. Intell. Transp. Syst.* 2022, 23, 12633–12647. [CrossRef]
- Arulmozhi, E.; Bhujel, A.; Moon, B.E.; Kim, H.T. The Application of Cameras in Precision Pig Farming: An Overview for Swine-Keeping Professionals. *Animals* 2021, 11, 2343. [CrossRef]
- 12. Yang, A.; Huang, H.; Yang, X.; Li, S.; Chen, C.; Gan, H.; Xue, Y. Automated video analysis of sow nursing behavior based on fully convolutional network and oriented optical flow. *Comput. Electron. Agric.* **2019**, *167*, 105048. [CrossRef]
- Chen, C.; Zhu, W.; Steibel, J.; Siegford, J.; Han, J.; Norton, T. Recognition of feeding behaviour of pigs and determination of feeding time of each pig by a video-based deep learning method. *Comput. Electron. Agric.* 2020, 176, 105642. [CrossRef]
- 14. Jiang, M.; Rao, Y.; Zhang, J.; Shen, Y. Automatic behavior recognition of group-housed goats using deep learning. *Comput. Electron. Agric.* **2020**, 177, 105706. [CrossRef]
- Gan, H.; Ou, M.; Huang, E.; Xu, C.; Li, S.; Li, J.; Liu, K.; Xue, Y. Automated detection and analysis of social behaviors among preweaning piglets using key point-based spatial and temporal features. *Comput. Electron. Agric.* 2021, 188, 106357. [CrossRef]
- 16. Alameer, A.; Kyriazakis, I.; Bacardit, J. Automated recognition of postures and drinking behaviour for the detection of compromised health in pigs. *Sci. Rep.* 2020, *10*, 13665. [CrossRef]
- 17. Psota, E.T.; Schmidt, T.; Mote, B.; Pérez, L.C. Long-Term Tracking of Group-Housed Livestock Using Keypoint Detection and MAP Estimation for Individual Animal Identification. *Sensors* **2020**, *20*, 3670. [CrossRef]
- Bhujel, A.; Arulmozhi, E.; Moon, B.E.; Kim, H.T. Deep-Learning-Based Automatic Monitoring of Pigs' Physico-Temporal Activities at Different Greenhouse Gas Concentrations. *Animals* 2021, *11*, 3089. [CrossRef]
- Zhang, L.; Gray, H.; Ye, X.; Collins, L.; Allinson, N. Automatic Individual Pig Detection and Tracking in Pig Farms. Sensors 2019, 19, 1188. [CrossRef]
- Tu, S.; Liu, H.; Li, J.; Huang, J.; Xue, Y. Instance Segmentation Based on Mask Scoring R-CNN for Group-housed Pigs. In Proceedings of the 2020 International Conference on Computer Engineering and Application (ICCEA), Guangzhou, China, 27–29 March 2020; pp. 458–462.
- Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
- Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN:Towards realtime object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 2017, 39, 1137–1149. [CrossRef]
- Law, H.; Deng, J. CornerNet: Detecting Objects as Paired Keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 734–750.
- Zhou, X.; Koltun, V.; Krähenbühl, P. Tracking Objects as Points. In European Conference on Computer Vision; Springer: Cham, Switzerland, 2020; pp. 474–490.
- 25. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. arXiv Prepr. 2021, arXiv:2107.08430.
- Luo, W.; Xing, J.; Milan, A.; Zhang, X.; Liu, W.; Kim, T.-K. Multiple object tracking: A literature review. Artificial Intelligence. *Artif. Intell.* 2021, 293, 103448. [CrossRef]
- Wang, Z.; Zheng, L.; Liu, Y.; Li, Y.; Wang, S. Towards Real-Time Multi-Object Tracking. In European Conference on Computer Vision; Springer: Cham, Switzerland, 2020; pp. 107–122. [CrossRef]
- 28. Sun, P.; Jiang, Y.; Zhang, R.; Xie, E.; Luo, P. TransTrack: Multiple-Object Tracking with Transformer. *arXiv Prepr.* 2020, arXiv:2012.15460.
- 29. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3464–3468. [CrossRef]
- Wojke, N.; Bewley, A.; Paulus, D. Simple online and realtime tracking with a deep association metric. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 3645–3649. [CrossRef]
- 31. Meinhardt, T.; Kirillov, A.; Leal-Taixe, L.; Feichtenhofer, C. TrackFormer: Multi-Object Tracking with Transformers. *arXiv Prepr.* **2021**, arXiv:2101.02702.
- 32. Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Cremers, D.; Reid, I.; Roth, S.; Schindler, K.; Leal-Taixé, L. MOT20: A benchmark for multi object tracking in crowded scenes. *arXiv Prepr.* **2020**, arXiv:2003.09003.
- Zhang, Y.; Wang, C.; Wang, X.; Zeng, W.; Liu, W. FairMOT: On the Fairness of Detection and Re-identification in Multiple Object Tracking. Int. J. Comput. Vis. 2021, 129, 3069–3087. [CrossRef]

- 34. Zhang, Y.; Sun, P.; Jiang, Y.; Yu, D. ByteTrack: Multi-Object Tracking by Associating Every Detection Box. *arXiv Prepr.* 2021, arXiv:2110.06864.
- 35. Wu, X.W.; Sahoo, D.; Hoi, S.C.H. Recent advances in deep learning for object detection. *Neurocomputing* **2020**, *396*, 39–64. [CrossRef]