sensors

MDPI

Article

Detecting Animal Contacts—A Deep Learning-Based Pig Detection and Tracking Approach for the Quantification of Social Contacts

Martin Wutke [1,2,*], Felix Heinrich [1], Pronaya Prosun Das [3], Anita Lange [2], Maria Gentz [2], Imke Traulsen [2], Friederike K. Warns [4], Armin Otto Schmitt [1,5] and Mehmet Gültas [5,6,*]

1 Breeding Informatics Group, Department of Animal Sciences, Georg-August University, Margarethe von Wrangell-Weg 7, 37075 Göttingen, Germany; felix.heinrich@uni-goettingen.de (F.H.); armin.schmitt@uni-goettingen.de (A.O.S.)
2 Livestock Systems, Department of Animal Sciences, Georg-August University, Albrecht-Thaer-Weg 3, 37075 Göttingen, Germany; anita.lange@agr.uni-goettingen.de (A.L.); maria.gentz@thuenen.de (M.G.); imke.traulsen@uni-goettingen.de (I.T.)
3 Bioinformatics Group, Fraunhofer Institute for Toxicology and Experimental Medicine (Fraunhofer ITEM), Nikolai-Fuchs-Str. 1, 30625 Hannover, Germany; pronaya.prosun.das@item.fraunhofer.de
4 Agricultural Test and Education Centre House Düsse, Chamber of Agriculture North Rhine-Westphalia, Haus Düsse 2, 59505 Bad Sassendorf, Germany; Friederike.Warns@LWK.NRW.DE
5 Center for Integrated Breeding Research (CiBreed), Georg-August University, Albrecht-Thaer-Weg 3, 37075 Göttingen, Germany
6 Statistics and Data Science, Faculty of Agriculture, South Westphalia University of Applied Sciences, 59494 Soest, Germany
* Correspondence: martin.wutke@uni-goettingen.de (M.W.); gueltas.mehmet@fh-swf.de (M.G.)

Abstract: The identification of social interactions is of fundamental importance for animal behavioral studies, addressing numerous problems like investigating the influence of social hierarchical structures or the drivers of agonistic behavioral disorders. However, the majority of previous studies often rely on manual determination of the number and types of social encounters by direct observation which requires a large amount of personnel and economical efforts. To overcome this limitation and increase research efficiency and, thus, contribute to animal welfare in the long term, we propose in this study a framework for the automated identification of social contacts. In this framework, we apply a convolutional neural network (CNN) to detect the location and orientation of pigs within a video and track their movement trajectories over a period of time using a Kalman filter (KF) algorithm. Based on the tracking information, we automatically identify social contacts in the form of head–head and head–tail contacts. Moreover, by using the individual animal IDs, we construct a network of social contacts as the final output. We evaluated the performance of our framework based on two distinct test sets for pig detection and tracking. Consequently, we achieved a Sensitivity, Precision, and F1-score of 94.2%, 95.4%, and 95.1%, respectively, and a *MOTA* score of 94.4%. The findings of this study demonstrate the effectiveness of our keypoint-based tracking-by-detection strategy and can be applied to enhance animal monitoring systems.

Keywords: pig detection; pig tracking; convolutional neural network; Kalman filter; precision livestock farming

Citation: Wutke, M.; Heinrich, F.; Das, P.P.; Lange, A.; Gentz, M.; Traulsen, I.; Warns, F.K.; Schmitt, A.O.; Gültas, M. Detecting Animal Contacts—A Deep Learning-Based Pig Detection and Tracking Approach for the Quantification of Social Contacts. *Sensors* **2021**, *21*, 7512. https://doi.org/10.3390/s21227512

Academic Editors: Dionysis Bochtis and Aristotelis C. Tagarakis

Received: 18 October 2021
Accepted: 10 November 2021
Published: 12 November 2021

1. Introduction

Today, it is well known that domestic pigs are highly social animals, maintaining hierarchical structures and socially organized groups. In commercial farming systems, the established social orders are frequently disrupted due to mixing groups as they are transferred between different housing and production stages [1,2]. Mixing of unacquainted animals leads to the establishment of a new social hierarchy going along with agonistic interactions which may result in reduced animal welfare and health [3–5].

In order to enhance animal welfare and health in future husbandry systems, the analysis of animal interactions as well as their monitoring and prediction is of high importance in research and commercial farming. Reasons for agonistic or aggressive behavior are manifold [2,6–10], contribute to a certain extent to the animal specific behavior, and also include a high variation between animals [2].

Nowadays, video recordings are a standard tool in research for observing pig pens due to their non-invasive nature. Recent technological advances including deep learning techniques, led to the rise of precision livestock farming applications to partially automate the time consuming video evaluation process [11]. Within the area of precision livestock farming, the tasks of multiple object detection and motion tracking have been studied intensively in recent years, in order to remotely monitor several animals and to capture the animals activity [11–15]. While multiple object detection refers to the task of locating several objects belonging to a category of interest within an image [16], multiple object tracking can be described as tracing the movement of objects throughout a consecutive number of video frames and consistently assigning individual object IDs [17].

With the recent advances in the area of deep learning, convolutional neural network (CNN)-based applications achieved state-of-the-art results in various image and video object detection scenarios [18]. Here, the most frequently used detection approaches aim to localize an object of interest by computing a bounding box around the object [19–21]. Although these approaches work successfully for various problem settings, due to the overlapping of the predicted bounding boxes, their applicability is limited for the analysis of videos with high utilization rates and several pigs in a close environment [22,23]. Furthermore, the standard bounding box approach only provides the positional information without taking the orientation of the animal into account which is a key information in order to reliably differentiate distinctive contact types like head–head interactions.

To overcome this limitation, an alternative detection approach was developed by Psota et al. [23]. The authors proposed a keypoint-based CNN for the detection of individual body parts of pigs. After processing the CNN output with a cross-check matching algorithm for assigning the individual body parts, they were able to successfully differentiate multiple animals even in a close proximity environment. Achieving a sensitivity, precision rate and an F1-score of 96%, 100%, and 98%, respectively, their approach proved to be highly successful in identifying the location and orientation of individual animals. As an extension, Psota et al. [11] applied this method to deal with the problem of tracking individual animals by using a second object detection CNN to detect ear tags which serve as a pig-individual identifier. Although this approach shows a lot of practical potential, using individual ear tags requires additional effort for the attachment of the ear-tags. Furthermore, the detectability of the corresponding ear-tags must be ensured, as the visibility of the tags is often prevented by heavy interactions between the animals, bad lighting conditions or a high degree of pollution in the pig compartment. As a possible solution, it seems beneficial to use the detected body parts directly for animal tracking which could increase the applicability of the keypoint-based detection approach, while simultaneously reducing the complexity without the need to train a second CNN for object detection.

Therefore, by following the idea in [23] we implement a CNN based framework for detecting individual body parts of pigs and use the predicted shoulder-tail information directly as the input for a Kalman filter (KF)-based tracking algorithm. The KF is currently one of the most frequently used approaches for tracking the motion activity of multiple objects within a video [24,25] which allows the assignment of individual animal IDs in this study. Subsequently, by collecting the shoulder-tail information as well as the animal ID in our framework we differentiate between specific head–head and head–tail contacts. As a result of this, our framework is able to determine a table of social contacts and to compute a graphical network of the social relationships. Such type of contacts could provide crucial information about social interactions including tail and ear biting. Consequently, using the proposed framework we aim to automate the process of video data analysis by quantifying the number of social encounters for several pigs within a video sequence. This information

can then be used by researchers to specifically analyze scenes of interest within their respective fields or to directly perform a SNA.

The remainder of this article is structured as follows. Section 2 describes the data used for this analysis and explains the methodical foundation as well as the evaluation rationale applied in this article. Next, the results for the animal detection, animal tracking, and social contact identification are presented and discussed in Section 3. Section 4 concludes this article.

2. Materials and Methods

In this section, the data used for the analysis as well as the different stages of the proposed method and the evaluation rationale are described in detail. The underlying multi-stage framework of the proposed method is illustrated in Figure 1.

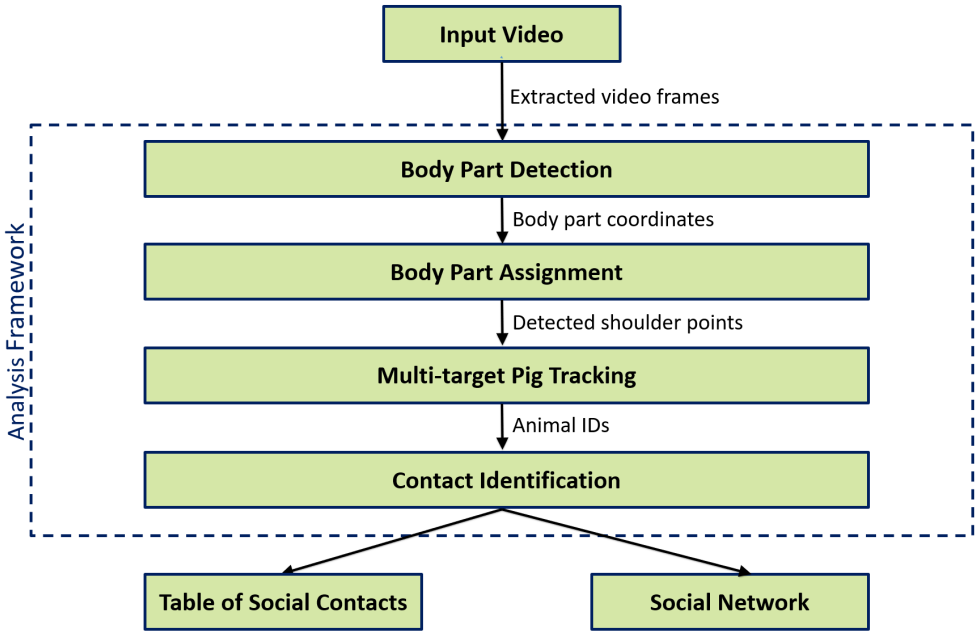

Figure 1. Flowchart of the analysis applied in this study.

The proposed method follows a tracking-by-detection (TBD) approach with the goal of tracking a known number of pigs within the pig compartment. As the input signal, a video sequence represented by a series of consecutive frames $S = (s_1, s_2, \ldots, s_N)$ is used, where N is the number of frames. In this context, TBD refers to first detecting objects of interest in each video frame using a pre-trained detector and then linking the independent detections at the temporal dimension over a longer period of frames [26,27].

In this study, the location and orientation of each pig within the video is determined frame-wise using a keypoint-based CNN to output the coordinates of important body parts (shoulder, tail, left ear, and right ear). After associating the body parts and assigning a unique ID to each pig, the shoulder coordinates in time T are used as the input signal for a Kalman filter to predict and track the location of future shoulder positions in $T + i$ with $i = (1, 2, \ldots, N - T)$. By tracking the shoulder points, two distinct types of body part contacts are identified as being either a head-head or head-tail contact. If two shoulder points are close to each other, the encounter is marked as a head-head contact. If a shoulder point is close to a tail point, the encounter is marked as a head–tail contact. Finally, by incorporating the frame number information, animal IDs and types of contact, a table of social contacts as well as a graphical representation in form of a social network is constructed as an output.

2.1. Data Acquisition and Processing

The video data used for this study was collected by [28,29] between October and December 2018 at the research farm Futterkamp of the Chamber of Agriculture of Schleswig-Holstein in Germany during a research project to investigate the effects of different farrowing and rearing systems on the stress level of piglets. For this purpose, a single static camera of the type AXIS M3024-LVE (Axis Communications AB, Lund, Sweden) was assembled 3 m above the ground which recorded all videos with a frame-per-second (fps) rate of 10 frames and a display resolution of 1280 × 800 pixels. For this study, sequences with a varying number of animals and a fixed camera angle have been selected. Figure 2 shows three example frames of structurally identical pens.

Figure 2. Example frames of the pig compartment under investigation with a known number of pigs.

In our analysis, we extracted all video frames and converted them to a grayscale format with a pixel dimension of 640 × 400 pixels, in order to avoid a potential bias of the CNN by differentiating between day and night recordings [30]. The dimensionality reduction was carried out to reduce CNN training time and, thereby, increase the computational efficiency.

2.2. Pig Detection

An essential step in the tracking of individual pigs is their successful detection. For this purpose, Psota et al. [23] established a keypoint-based CNN to detect distinct body parts of pigs and highlighted the advantage of this approach over existing bounding box detections. Using this keypoint approach, we implemented a CNN to receive a video frame and to output the coordinates of four individual body parts for each animal. Similar to the work in [23], we stored the coordinate information of the left ear, right ear, shoulder and tail point directly as a binary image in a separate image channel (Figure 3B–E). Additionally, the information for the connection lines shoulder–tail, shoulder–left ear, and the shoulder–right ear are included (Figure 3F–H). In comparison to conventional top-down detection methods, which output bounding box or ellipsoid coordinates, the detected keypoints directly provide a pose representation which facilitates the contact identification of the animals [23].

During the training process, a CNN is trained to map the input image to the ground truth annotations by highlighting the important pixels of the corresponding body parts. The architecture of our CNN follows an autoencoder structure which is illustrated in Figure 4.

The CNN consists of 25 convolutional layers combined with 2 × 2 max pooling and upsampling layers. The first ten layers are used to reduce the input dimension from 640 × 400 pixels to a latent representation of 40 × 25 pixels and extract the main features for the body part detection. At the lowest dimension, six stacked convolutional layers forward the latent image representation to a set of upsampling and convolutional layers, which step-wise increase the image dimension back to 640 × 400 pixels and output the approximate body part coordinates. After each upsampling layer, a residual connection with a concatenation layer is used to copy a representation from the encoder layers to the decoder layers to decrease the reconstruction loss and improve the training efficiency [23,31]. All convolutional layers are implemented with a ReLU activation function, zero padding and a stride parameter of 1. Using the Adam optimizer [32] and binary cross-entropy loss, the

network applies a sigmoid activation for the last layer to output pixel intensities between 0 and 1.

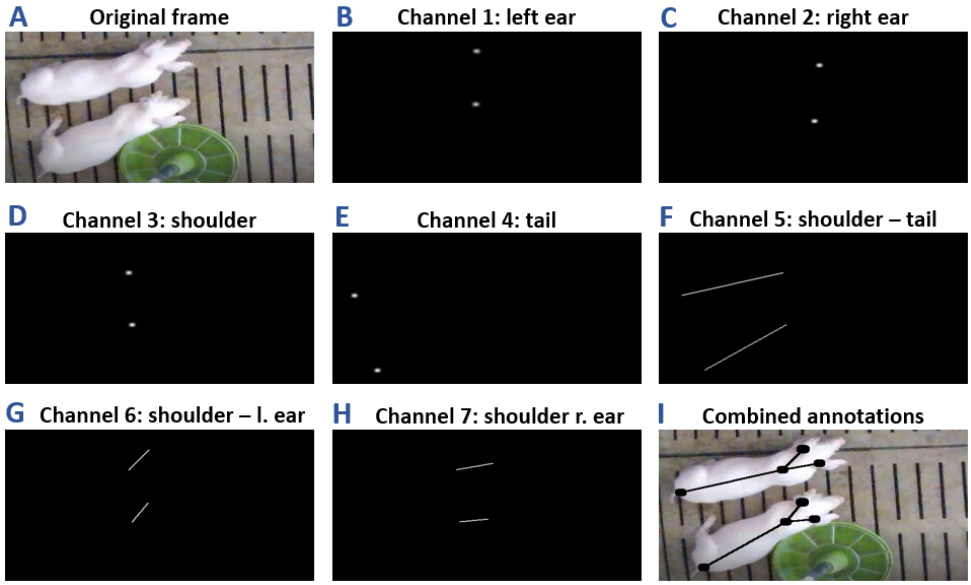

Figure 3. (**A**) The original image which serves as the input for the CNN. (**B**–**H**) The corresponding ground truth annotations containing the positional body part information which are used for the training process. (**I**) The original image combined with the ground truth annotations.

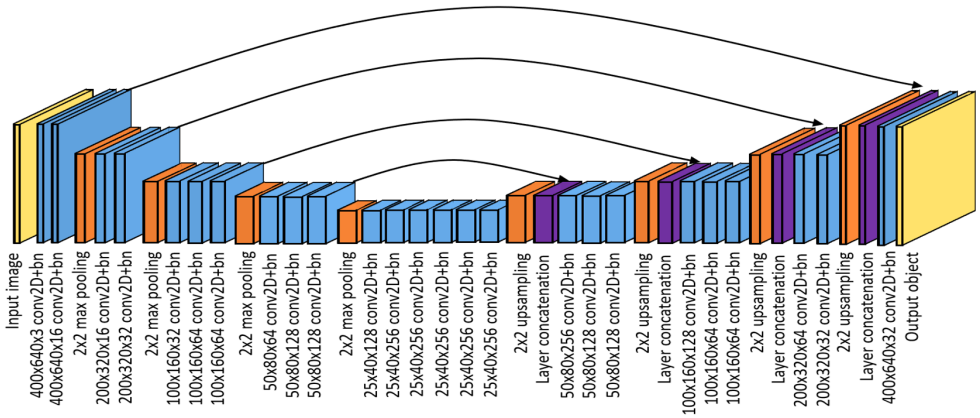

Figure 4. The implemented CNN follows an autoencoder structure to create the seven-channel output object given a gray-scaled video frame. For each convolutional layer the dimensional information is given in the format height × width × number of convolutional filters.

The CNN was implemented in Python (version 3.7.6) [33] using the deep learning framework Keras (version 2.2.4) [34] with TensorFlow (version 2.0) [35] as a backend. The model training was carried out on a workstation equipped with two Intel Xeon Gold 6138 CPUs, 512 GB RAM, and a NVIDIA Quadro P5000 GPU.

Subsequently, to train the CNN, we annotated a training data set consisting of 2457 images. To increase the overall sample size and to enable the model to see more heterogenous animal postures, we augmented the training images as well as the corresponding ground truth annotations using vertical and horizontal shifting, shrinking and image rotation. After augmentation, the total training data set had a size of 12,285 images of which 90% were used for training and 10% for the model validation after each epoch.

After the CNN has learned to predict the positions of the individual body parts in the two-dimensional image space, the location and orientation of each pig are determined based on the CNN output. For this step, we mainly focus on the analysis of Channel 5

(Figure 3F), as it mainly carried the most accurate and robust information. By extracting the start and end coordinates of each shoulder-tail line, a depth-first-search algorithm (DFS) [36] is applied to determine the pigs location. However, Channel 5 does not contain the information of the animal's orientation. Therefore, we incorporate the information of Channels 1–4 to identify shoulder and tail points of each pig. An example frame after the detection process is given in Figure 5.

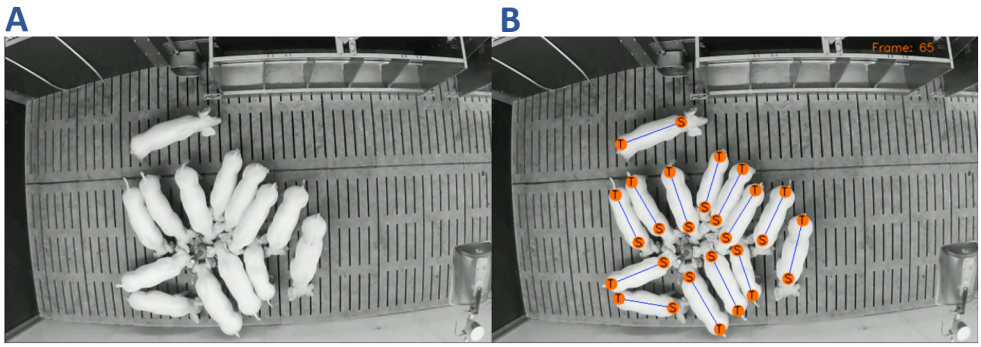

Figure 5. Example of the CNN pig detection showing the original frame (**A**) and the detected shoulder and tail points (**B**). The frame shows a feeding situation in which several pigs are in close proximity to each other. Each pig is marked by highlighting the corresponding shoulder and tail points as well as the connection line.

2.3. Pig Tracking

After determining the location and orientation of each pig, this step aims to track the pigs' movement and to link their trajectories over the total sequence of frames. While previous approaches mainly focused on pig tracking using bounding box detections [19,37–39], the suitability of these approaches is limited to identify social contacts in close proximities because they do not incorporate the animals' orientation and show a high risk of ID switches in situations of overlapping boxes. To reduce this limitation, we apply for the first time a combination of a KF tracking approach [40] with the body part detections as the input signal to track individual animals and to determine distinct contact types. While the CNN output can still contain false detections, we use the KF as an unsupervised, dynamic model to estimate and track the shoulder positions, even in frames in which the true point could not be detected by the CNN.

The KF process is divided into two phases: a prediction and an update phase. In the prediction phase, a prediction of the shoulder position for the current time step k is computed based on the KF estimate of the shoulder point of the previous time step $k-1$. During the update phase, new CNN detections at k are used to adjust the current prediction and to compute the KF estimate at time k, which is used as new input for the prediction phase of the next time step $k+1$. In the case of false positive or false negative CNN detections, the input signal variance is increased, which leads the KF to weight down the importance of the CNN input and to increase the weight of the previous KF prediction [41]. Consequently, the KF yields a more robust estimate of the shoulder point coordinates which overcomes the problem of shoulder–tail swaps and misdetections.

After the KF is initialized, it is applied to the ordered shoulder points produced by the CNN. While all videos have been recorded with a fps rate of 10 frames, even intense motion changes of the animals caused just slight pixel variations in the consecutive frames. Therefore, a KF shoulder point in frame k is mapped to the corresponding KF shoulder point in frame $k+1$ by minimizing the Euclidean distance. Consequently, each KF shoulder estimate is assigned an individual animal ID. An example of the pig tracking and ID assignment is shown in Figure 6.

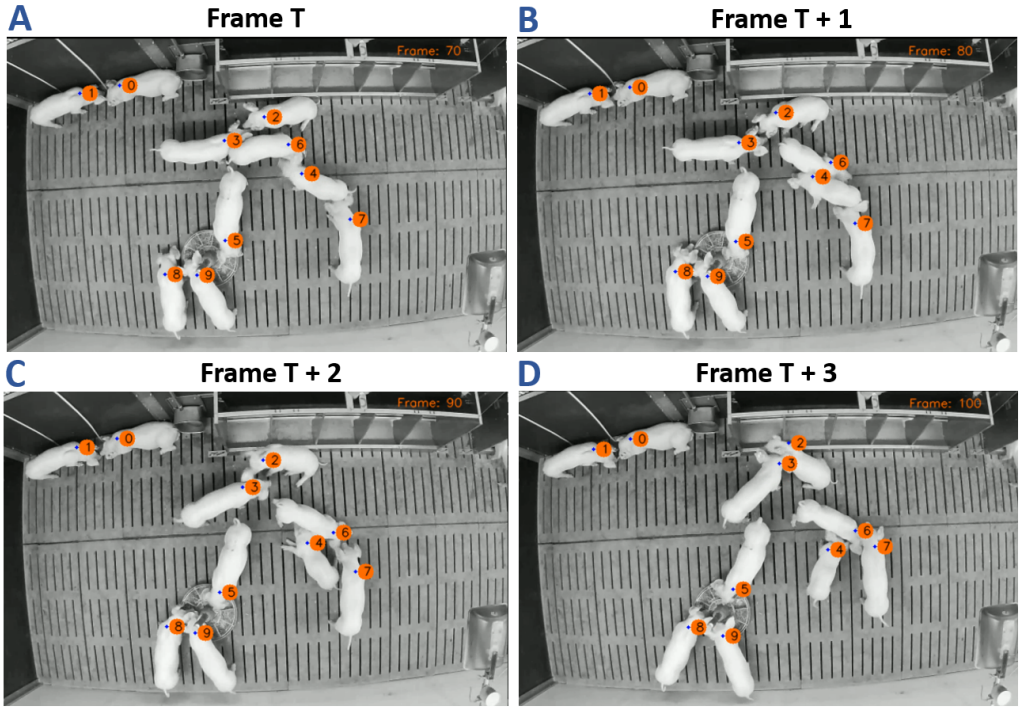

Figure 6. The multi-target pig tracking, shown for an example sequence of four frames, on a one second interval. The frame number is printed on the top, and the fps-rate was set to ten frames. The KF estimate of the shoulder point is highlighted by the blue dot, near the shoulder region of each pig. The corresponding ID is placed right of it.

2.4. Identifying Contact Information

The last stage of our proposed framework aims to identify animal contacts in the form of either head–head or head–tail contacts, which may be related to tail biting or nosing behavior. We use the KF shoulder and tail estimates, as well as the pig orientation to define a region of interest at the head and tail area of each pig. If at least two pigs are nearby to each other so that either both head regions or one head and one tail region are sufficiently close, the head and tail regions are intersecting, which indicates a potential contact. To account for different age and size levels of the animals, the average length of each shoulder–tail line per frame is calculated and used to scale the head and tail regions to a radius r, defined as:

$$r = \frac{1}{\alpha N} \sum_{i=1}^{N} \sqrt{(s_i - t_i)^2} \quad (1)$$

where s_i and t_i are the shoulder and tail coordinates of the i-th pig, N the total number of pigs in the given frame, and α a scaling factor. In this study, an α value of 3 was empirically deemed to be optimal for computing an area of interest large enough to cover the essential part of the head and tail region, but being small enough to avoid potential false detections in the form of animals walking by. Figure 7 shows an example of the region computation and contact identification.

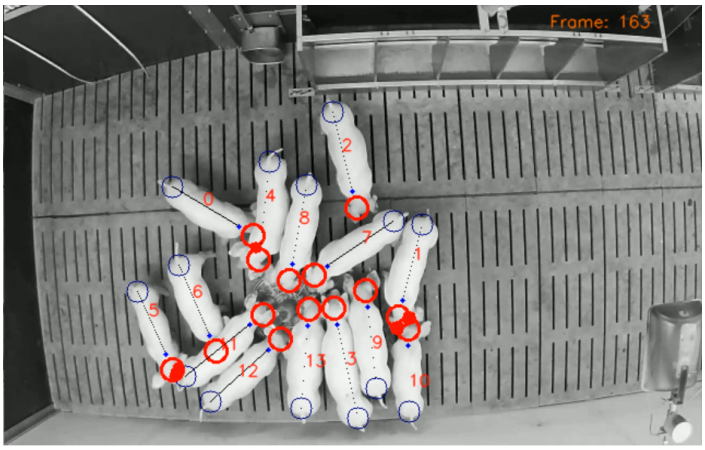

Figure 7. Based on the KF estimates, a region of interest for each head and tail area is computed. By detecting the intersection of at least two regions, the type of contact and the associated animals can be identified. Exemplarily, a head–tail contact can be detected between pigs 5 and 11 and a head–head contact for pigs 1 and 10.

2.5. Pig Detection and Tracking Evaluation Rationale

In order to assess the overall performance of our framework, we evaluated both the CNN detection stage as well as the multi-target pig tracking stage separately. For the CNN detection, we additionally annotated 100 randomly selected images and used these frames as a test set to evaluate the CNN's ability to predict the location of individual pigs by detecting their shoulder points. To avoid confusion we used the subscript "D" and "T" to differentiate between the detection evaluation and the tracking evaluation. For the detection stage, we computed the number of *True Positives* (TP_D), *False Positives* (FP_D), and *False Negatives* (FN_D) over all test images and calculated the *Sensitivity*, *Precision* rate, and *F1-score* defined as

$$Sensitivity = \frac{TP_D}{P_D} \tag{2}$$

$$Precision = \frac{TP_D}{TP_D + FP_D} \tag{3}$$

$$F1 = \frac{2TP_D}{2TP_D + FP_D + FN_D} \tag{4}$$

In order to determine TP_D, FP_D, and FN_D, a circular detection region around the true shoulder point was defined by computing the average distance from the true shoulder point to the true left and right ear points over all pigs of the given frame as the radius. If exactly one shoulder point was predicted by the CNN within the detection region, this point has been classified as TP_D. If more than one point was predicted within the region or outside the detection region, these points have been classified as FP_D. If no point was detected within the region, the point was classified as FN_D.

Despite the recent advances of multiple object tracking applications, there is still a lack of large-scale benchmarks and comparable evaluation metrics [42,43]. While the majority of existing object tracking publications applies a bounding box approach using the intersection over union (IoU) as an evaluation criterion between the annotated and predicted box, our proposed framework applies a keypoint-based approach, for which the IoU is not suitable. Therefore, we followed previous studies [21,38,44] and calculated the *Multiple Object Tracking Accuracy* (*MOTA*) defined as

$$MOTA = 1 - \frac{FP_T + FN_T + IDSW}{N_{CNN}} \tag{5}$$

by manually determining the number of falsely tracked pigs (FP_T), pigs which have not been tracked (FN_T), the number of ID switches ($IDSW$), and the number of pigs detected

by the CNN (N_{CNN}). While the total tracking error can be further divided into detection errors, association errors, and localization errors [42,45], FP_T and FN_T account for the detection errors and $IDSW$ accounts for the association errors. If a detected pig is not tracked by the KF tracker, it is classified as FN_T. If the tracker tracks something different than a pig, it is classified as an FP_T. We further increased the number of FP_T to account for the localization errors if a pig is tracked, but the corresponding tracking point is too far away from the target point. To determine the $MOTA$ value we randomly selected 70 videos as test sequences with an average length of 20 seconds. These sequences have been analyzed by the CNN in advance, in order to obtain the coordinates of the detected body parts.

3. Results and Discussion

By applying our pig detection and tracking framework, we first analyzed in this study several video frames including different scenarios, in order to assess the detection and tracking performance. After that, three distinct animal contact network visualizations are presented and discussed to demonstrate the applicability and functionality of our framework.

3.1. Pig Detection and Tracking

As the performance of a tracking-by-detection (TBD) algorithm depends strongly on the accuracy of the corresponding detector [46], we evaluated the CNN performance for locating the pig position and determining its orientation using the *Sensitivity*, *Precision*, and *F1-Score* metric, respectively. For this purpose, we analyzed the manually annotated detection test set containing 100 randomly selected frames. Consequently, in total 1054 shoulder points have been annotated and considered for the evaluation analysis (see Table 1). In a following step, we focused on the tracking ability of the implemented KF and used 70 randomly selected video sequences as the tracking evaluation data. The results for the detection as well as for the tracking are provided in Table 1. The data sets used for the detection and tracking evaluation are made publicly available at https://github.com/MartinWut/Supp_DetAnIn (accessed on 9 November 2021).

Table 1. Evaluation results for the pig detection set and tracking set.

Test Set	TP	FP	FN	IDSW	Sensitivity (%)	Precision (%)	F1 (%)	MOTA (%)
Detection	1019	51	35	-	94.2	95.4	95.1	-
Tracking	640	20	8	10	-	-	-	94.4

Table 1 shows that the majority of shoulder points was successfully detected by the CNN resulting in high performance values, in terms of *Sensitivity*, *Precision*, and *F1-Score*. While in total 1054 shoulder points have been manually annotated, only a relatively small fraction of these points have not been detected. On the other hand, the number of falsely detected pigs (FP = 51) indicates that the detection CNN still has limitations in challenging situations like object occlusion. Figure 8 illustrates several example frames from the test set showing cases of successful and failed detections.

In Figure 8, it can be observed that the CNN successfully detected most of the pigs. Focusing on the failed detections, the large majority of failures was produced in situations in which one pig has been occluded by another pig, which led to a false negative detection. However, the problem of object occlusion and the resulting degradation in performance is not linked to the design of this study. In fact, the issue of object detection under the influence of occlusion is a challenging task which negatively affects the robustness of most detection algorithms [47]. While current approaches aim to tackle this problem by applying a compositional neural network structure in combination with an occluder model [47–50], the majority of approaches focus on the problem of partial occlusion and would, therefore, be of limited suitability for this study. Moreover, during the tracking process, the negative effect of object occlusion can be reduced to some extent, by applying a predictive model like

the KF algorithm, which internally interprets the CNN detections as noisy measurement information. In cases of extreme volatility, like the loss of an object due to occlusion, the KF reduces the influence of the measurement input by increasing the importance of the Kalman prediction [51,52]. To reduce the effect of a missing shoulder or tail point to some extent, we computed the number of detected objects frame-wise and marked the corresponding frames as corrupted in cases of missing detections. Corrupted frames have then been excluded for the KF tracking where we used the previous KF estimates as the new measurement input instead.

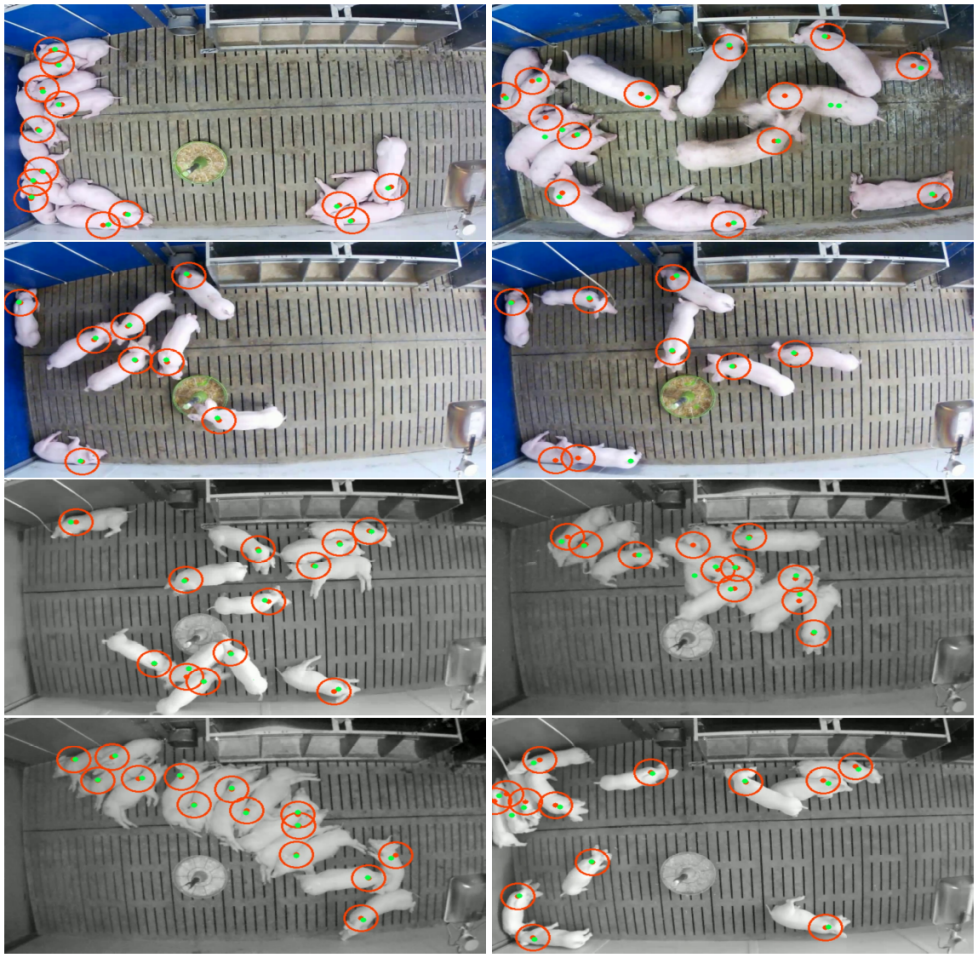

Figure 8. Example frames from the detection test set for day and night frames showing cases of a completely successful pig detection (**left column**) and cases in which at least one pig was not detected correctly (**right column**). A true shoulder point is highlighted by a red dot in the middle of the detection region (red circle). A green dot represents the predicted shoulder point from the CNN.

To assess the tracking performance, we further analyzed 70 test sequences containing various scenarios like feeding, resting, or interactions. As it can be seen in Table 1, the implemented KF was able to track 640 out of 678 shoulder points correctly resulting in a *MOTA* score of 94.4%. However, in 38 cases the tracking of the detected shoulder points failed: (i) 20 animals have not been tracked, (ii) eight animals have been tracked at the wrong position, and (iii) ten cases occurred in which the assigned ID of two pigs switched. Examples of a *FP*-track, a *FN*-track, and an ID switch are given in Figure 9.

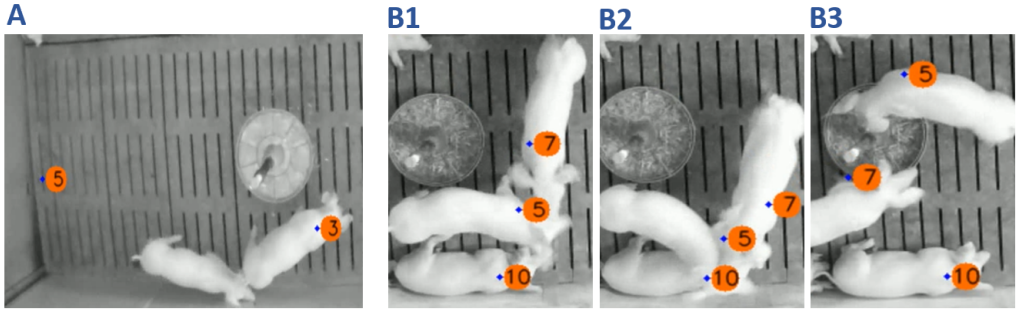

Figure 9. Examples of a failed tracking performance. (**A**) While the shoulder point of the lower pig was not tracked (*FN*), the corresponding ID was computed at an empty spot in the compartment (*FP*). (**B1**–**B3**) A case of an ID switch due to a close proximity interaction and the occlusion of an animal. Before the interaction takes place, ID 5 and ID 7 have been assigned correctly (**B1**). After the interaction the IDs switched so that the ID 5 was assigned ID 7 and vice versa (**B3**).

In line with previous studies [26], we observed that if the detector is able to detect all pigs correctly, the corresponding pig tracking is working without producing corrupted tracks. In contrast, if the detection of the body parts fails, the tracker predicts the shoulder point based on the movement pattern of the previous frames. While the consequences are minor for short periods of detection failures, longer phases of missing detections lead to the effect of misplaced tracking IDs. However, this limitation is not specific to this study. Although the most successful tracking approaches are based on a TBD-strategy, the consequence of missing detections can be a significant reduction in their performance [53]. An example of a misplaced tracking ID is given in Figure 9A.

Another fundamental issue in the field of multiple object tracking, is the problem of ID switches, which is shown in Figure 9(B1–B3) [21,39,44,54]. In Figure 9(B1), the pigs with IDs 5, 7, and 10 are successfully detected and tracked. During the sequence, pigs 5 and 7 are occluding the shoulder point of pig 10, which leads to a missed detection of this animal by the detector (Figure 9(B2)). However, after all shoulder points reappear in the video, the KF tracker estimates the position of the shoulder points, but swaps IDs 5 and 7 (Figure 9(B3)).

Unlike previous bounding box-based studies, the problem of ID switches in our study only occurs in cases when the shoulder points of the animals disappear due to different obstacles, thus preventing the detection of these points. While several existing bounding box tracking applications suffer from ID switches arising from highly overlapping boxes [39], the keypoint-based approach applied in this study considers a much smaller tracking area. Therefore, a strong overlap of two or more tracked shoulder points is less likely to occur, which explains the relatively low number of ID switches given in Table 1. Of particular interest, we further studied all cases of ID switches in the test set in order to establish the main reason for the ID switch issue. We found that nine out of ten ID switches have been caused by a missed detection rather than by two detected IDs in a short distance. Only in one case a false detection caused the ID switch.

3.2. Animal-to-Animal Contact Identification

The knowledge about social interactions is fundamental to enhance farming conditions and animal welfare. Thus, the final stage of the framework proposed in this study aims to identify such behavior patterns based on pig interactions. In particular, by focusing on close proximity contacts between at least two animals, our framework automatically takes into account head–head as well as head–tail contacts. After that, based on these contact information, we construct a trajectory map to highlight individual movement patterns, which finally provides an information table about the social contacts. An example of the identification process for one test sequence is given in Figure 10.

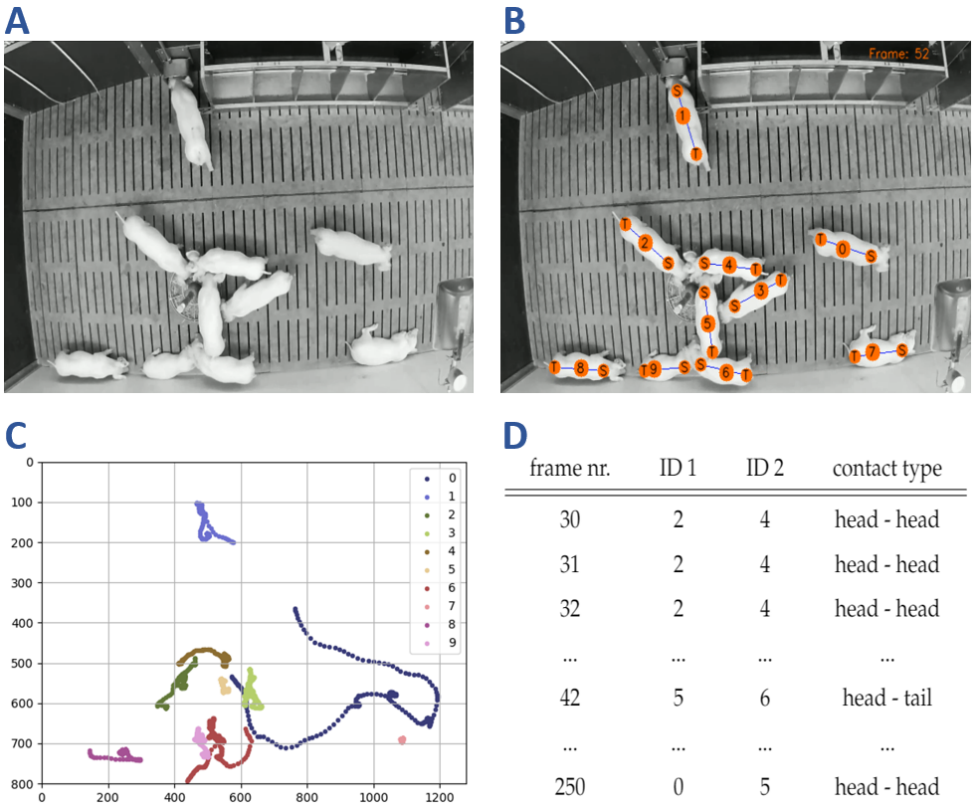

Figure 10. The social contact identification starts with a raw video frame (**A**). After detecting the shoulder and tail position (**B**), the trajectories are computed and analyzed over time (**C**). By identifying cases of close distances, a table of social contacts is constructed automatically (**D**).

The table of social contacts (Figure 10D) contains highly essential information about the contact pattern of animals over all video frames. As Smith et al. [55] pointed out, these data are crucial in the field of behavioral ecology and the automatic contact identification can reduce observer bias as a limiting factor. To address different research questions like the investigation of hierarchical structures, agonistic behavior patterns or pen utilization, the necessary information can be derived from this table as required. Further, this table can be used for extracting a distinct contact period by restricting the frame and ID information. However, the aim of this study is to differentiate between individual contacts. Therefore, the extraction process primarily focuses on the contact type information which is used to visualize the social relationships of the observed animals. Figure 11 shows an example for the visualization depicting both type of contacts (Figure 11A) as well as the specific head–head and head–tail contact types (Figures 11B,C).

Each of the three social networks in Figure 11 is constructed by using the individual animal ID as the node information and the contact frequency as an intensity score for the edge weight. In particular, for a holistic analysis, the consideration of all contact types is crucial to establish the general contact patterns between all animals in a compartment (Figure 11A). On the other hand, with regard to specific research questions [56–58], a more differentiated network design focusing on a distinct type of contact can be advantageous (Figures 11B,C). As a result, the structure of the specific network types differs from the holistic network which arises from the alterations in the edge weights.

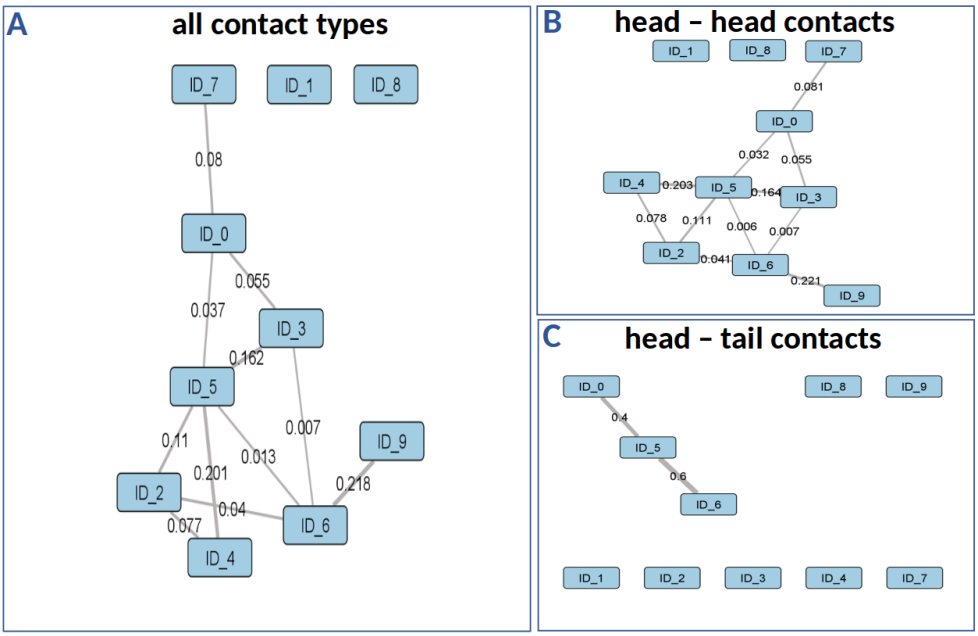

Figure 11. Examples of three social networks from one test sequence.

In comparison to previous studies which aim to tackle the issue of identifying social interactions by following a bounding box approach [59], our strategy is not restricted to specific behavioral patterns like escaping and chasing motion activities. Even in challenging situations like resting behavior where most of the pigs are lying down in a very small area, our proposed framework is able to identify social contacts to a certain extent.

To further extend the performance of the proposed framework, future work could focus on the implementation of more sophisticated network topologies. In this regard, recent studies [60,61] successfully showed the potential of attention mechanisms, introduced by Bahdanau et al. [62], to leverage the power of deep learning for highlighting important features. Ghaffarian et al. [60] performed a literature review analyzing 176 articles focusing on image classification, object detection, and change detection. As a result, the authors concluded that the majority of deep learning-based research studies reported a performance increase when applying an attention mechanism. This improvement could have the potential to further reduce the number of false positive predicted body parts and could enable additional filtering steps to detect corrupted video frames.

4. Conclusions

Today, the usage of video technology for animal monitoring is well established. However, extracting useful information is often challenging and thus limiting the potential of animal video analysis. In this study, we propose a framework for the automatic detection of social contacts to address the limitations of animal behavioral studies. By applying a keypoint-based body part detection and a subsequent pig tracking algorithm, we are able to determine the time, the animals involved, and the type of a social contact. We further process the information to construct a social network based on the contact type. To the best of our knowledge, this is the first study incorporating both a body part detection CNN as well as a Kalman filter tracking algorithm to identify social contacts. Our findings show the applicability of our approach to monitor a known number of pigs which can be used as part of early warning systems for the detection of behavioral changes. Overall, we suggest that our framework is applicable for different livestock animal monitoring systems.

Author Contributions: M.G. (Mehmet Gültas), I.T. and A.O.S. supervised the research. M.W. developed the model together with M.G. (Mehmet Gültas) and participated in the design of the study, prepared the data sets, conducted the analyses, and implemented the framework. F.H., P.P.D. and I.T. were involved in the interpretation of the results, together with M.W. and M.G. (Mehmet Gültas). A.L., F.K.W. and M.G. (Maria Gentz) provided the video data. M.W. and M.G. (Mehmet Gültas) wrote the final version of the manuscript. M.G. (Mehmet Gültas), A.O.S. and I.T. conceived the study. M.G. (Mehmet Gültas) designed the study and managed the project. All authors have read and agreed to the published version of the manuscript.

Funding: The project (InnoPig) was supported by funds of the German Government's Special Purpose Fund deposited at the Landwirtschaftliche Rentenbank (project no.: 2817205413; 758914).

Institutional Review Board Statement: The authors declare that the experiment was in accordance with current German law. As such, no part of this research was subject to approval of an ethics committee.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data sets used for the detection and tracking evaluation are made publicly available at https://github.com/MartinWut/Supp_DetAnIn (accessed on 9 November 2021).

Acknowledgments: We thank A. Rajavel (Breeding Informatics Group, University of Göttingen) for proofreading and her valuable advice. We thank the Chamber of Agriculture of Schleswig-Holstein for their support in data acquisition. We acknowledge support by the German Research Foundation and the Open Access Publication Funds of the Göttingen University.

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

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional neural network
KF	Kalman filter
SNA	Social network analysis
TBD	Tracking-by-detection

References

1. Verdon, M.; Rault, J.L. Aggression in group housed sows and fattening pigs. In *Advances in Pig Welfare*; Woodhead Publishing: Oxford, UK, 2018; pp. 235–260.
2. Foister, S.; Doeschl-Wilson, A.; Roehe, R.; Arnott, G.; Boyle, L.; Turner, S. Social network properties predict chronic aggression in commercial pig systems. *PLoS ONE* **2018**, *13*, e0205122. [CrossRef] [PubMed]
3. Büttner, K.; Scheffler, K.; Czycholl, I.; Krieter, J. Social network analysis-centrality parameters and individual network positions of agonistic behavior in pigs over three different age levels. *Springerplus* **2015**, *4*, 185. [CrossRef] [PubMed]
4. Rhim, S.J.; Son, S.H.; Hwang, H.S.; Lee, J.K.; Hong, J.K. Effects of mixing on the aggressive behavior of commercially housed pigs. *Asian-Australas. J. Anim. Sci.* **2015**, *28*, 1038. [CrossRef] [PubMed]
5. Stukenborg, A.; Traulsen, I.; Puppe, B.; Presuhn, U.; Krieter, J. Agonistic behaviour after mixing in pigs under commercial farm conditions. *Appl. Anim. Behav. Sci.* **2011**, *129*, 28–35. [CrossRef]
6. Morrone, B.; Bernardino, T.; Tatemoto, P.; Rodrigues, F.A.M.L.; de Moraes, J.E.; da Cruz, T.D.A.; Zanella, A.J. Indication that the presence of older conspecifics reduces agonistic behaviour in piglets at weaning. *Appl. Anim. Behav. Sci.* **2021**, *234*, 105201. [CrossRef]
7. Camerlink, I.; Proßegger, C.; Kubala, D.; Galunder, K.; Rault, J.L. Keeping littermates together instead of social mixing benefits pig social behaviour and growth post-weaning. *Appl. Anim. Behav. Sci.* **2021**, *235*, 105230. [CrossRef]
8. Marinelli, L.; Mongillo, P.; Carnier, P.; Schiavon, S.; Gallo, L. A Short Period of Darkness after Mixing of Growing Pigs Intended for PDO Hams Production Reduces Skin Lesions. *Animals* **2020**, *10*, 1729. [CrossRef] [PubMed]
9. Brajon, S.; Ahloy-Dallaire, J.; Devillers, N.; Guay, F. The role of genetic selection on agonistic behavior and welfare of gestating sows housed in large semi-static groups. *Animals* **2020**, *10*, 2299. [CrossRef]
10. Weller, J.E.; Camerlink, I.; Turner, S.P.; Farish, M.; Arnott, G. Socialisation and its effect on play behaviour and aggression in the domestic pig (Sus scrofa). *Sci. Rep.* **2019**, *9*, 4180. [CrossRef]
11. Psota, E.; 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] [PubMed]

12. Li, G.; Huang, Y.; Chen, Z.; Chesser, G.D.; Purswell, J.L.; Linhoss, J.; Zhao, Y. Practices and Applications of Convolutional Neural Network-Based Computer Vision Systems in Animal Farming: A Review. *Sensors* **2021**, *21*, 1492. [CrossRef]
13. Liu, C.; Zhou, H.; Cao, J.; Guo, X.; Su, J.; Wang, L.; Lu, S.; Li, L. Behavior Trajectory Tracking of Piglets Based on DLC-KPCA. *Agriculture* **2021**, *11*, 843. [CrossRef]
14. Matthews, S.G.; Miller, A.L.; Clapp, J.; Plötz, T.; Kyriazakis, I. Early detection of health and welfare compromises through automated detection of behavioural changes in pigs. *Vet. J.* **2016**, *217*, 43–51. [CrossRef] [PubMed]
15. Brünger, J.; Traulsen, I.; Koch, R. Randomized global optimization for robust pose estimation of multiple targets in image sequences. *Math. Model. Comput. Methods* **2015**, *2*, 45–53.
16. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
17. Kale, K.; Pawar, S.; Dhulekar, P. Moving object tracking using optical flow and motion vector estimation. In Proceedings of the 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), Noida, India, 2–4 September 2015; pp. 1–6.
18. Padilla, R.; Netto, S.L.; da Silva, E.A. A survey on performance metrics for object-detection algorithms. In Proceedings of the 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), Niteroi, Brazil, 1–3 July 2020; pp. 237–242.
19. Van Der Zande, L.; Guzhva, O.; Rodenburg, T.B. Individual detection and tracking of group housed pigs in their home pen using computer vision. *Front. Anim. Sci.* **2021**, *2*, 10. [CrossRef]
20. Ahn, H.; Son, S.; Kim, H.; Lee, S.; Chung, Y.; Park, D. EnsemblePigDet: Ensemble Deep Learning for Accurate Pig Detection. *Appl. Sci.* **2021**, *11*, 5577. [CrossRef]
21. Zhang, L.; Gray, H.; Ye, X.; Collins, L.; Allinson, N. Automatic individual pig detection and tracking in pig farms. *Sensors* **2019**, *19*, 1188. [CrossRef]
22. Steffen Küster, P.; Nolte, C.; Meckbach, B.; Stock, I.; Traulsen, I. Automatic behavior and posture detection of sows in loose farrowing pens based on 2D-video images. *Front. Anim. Sci.* **2021**, *2*, 23.
23. Psota, E.T.; Mittek, M.; Pérez, L.C.; Schmidt, T.; Mote, B. Multi-pig part detection and association with a fully-convolutional network. *Sensors* **2019**, *19*, 852. [CrossRef]
24. Madhukar, P.S.; Prasad, L. State Estimation using Extended Kalman Filter and Unscented Kalman Filter. In Proceedings of the 2020 International Conference on Emerging Trends in Communication, Control and Computing (ICONC3), Lakshmangarh, India, 21–22 February 2020; pp. 1–4.
25. Corrales, J.A.; Candelas, F.; Torres, F. Hybrid tracking of human operators using IMU/UWB data fusion by a Kalman filter. In Proceedings of the 2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI), Amsterdam, The Netherlands, 12–15 March 2008; pp. 193–200.
26. Sun, Z.; Chen, J.; Chao, L.; Ruan, W.; Mukherjee, M. A survey of multiple pedestrian tracking based on tracking-by-detection framework. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 1819–1833. [CrossRef]
27. Bogun, I.; Ribeiro, E. Robstruck: Improving occlusion handling of structured tracking-by-detection using robust kalman filter. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3479–3483.
28. Lange, A.; Gentz, M.; Hahne, M.; Lambertz, C.; Gauly, M.; Burfeind, O.; Traulsen, I. Effects of different farrowing and rearing systems on post-weaning stress in piglets. *Agriculture* **2020**, *10*, 230. [CrossRef]
29. Gentz, M.; Lange, A.; Zeidler, S.; Lambertz, C.; Gauly, M.; Burfeind, O.; Traulsen, I. Tail lesions and losses of docked and undocked pigs in different farrowing and rearing systems. *Agriculture* **2020**, *10*, 130. [CrossRef]
30. Wutke, M.; Schmitt, A.O.; Traulsen, I.; Gültas, M. Investigation of Pig Activity Based on Video Data and Semi-Supervised Neural Networks. *AgriEngineering* **2020**, *2*, 581–595. [CrossRef]
31. Deepak, K.; Chandrakala, S.; Mohan, C.K. Residual spatiotemporal autoencoder for unsupervised video anomaly detection. *Signal Image Video Process.* **2021**, *15*, 215–222. [CrossRef]
32. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
33. Rossum, G.V. Python Software Foundation. Python Language Reference, Version 3.7. 1995. Available online: http://www.python.org (accessed on 9 November 2021).
34. Chollet, F. Keras. 2015. Available online: https://keras.io (accessed on 9 November 2021).
35. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. Available online: https://www.tensorflow.org/ (accessed on 9 November 2021).
36. Spampinato, D.G.; Sridhar, U.; Low, T.M. Linear algebraic depth-first search. In Proceedings of the 6th ACM SIGPLAN International Workshop on Libraries, Languages and Compilers for Array Programming, Phoenix, AZ, USA, 22 June 2019; pp. 93–104.
37. Sun, L.; Li, Y. Multi-target pig tracking algorithm based on joint probability data association and particle filter. *Int. J. Agric. and Biol. Eng.* **2021**, *14*, 199–207. [CrossRef]
38. Gan, H.; Ou, M.; Zhao, F.; Xu, C.; Li, S.; Chen, C.; Xue, Y. Automated piglet tracking using a single convolutional neural network. *Biosyst. Eng.* **2021**, *205*, 48–63. [CrossRef]

39. Bochinski, E.; Senst, T.; Sikora, T. Extending IOU based multi-object tracking by visual information. In Proceedings of the 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Auckland, New Zealand, 27–30 November 2018; pp. 1–6.
40. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45. [CrossRef]
41. Welch, G.; Bishop, G. *An Introduction to the Kalman Filter*; University of North Carolina, Department of Computer Science: Chapel Hill, NC, USA, 1995.
42. Luiten, J.; Osep, A.; Dendorfer, P.; Torr, P.; Geiger, A.; Leal-Taixé, L.; Leibe, B. Hota: A higher order metric for evaluating multi-object tracking. *Int. J. Comput. Vis.* **2021**, *129*, 548–578. [CrossRef]
43. Fan, H.; Bai, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Huang, M.; Liu, J.; Xu, Y.; et al. Lasot: A high-quality large-scale single object tracking benchmark. *Int. J. Comput. Vis.* **2021**, *129*, 439–461. [CrossRef]
44. Cowton, J.; Kyriazakis, I.; Bacardit, J. Automated individual pig localisation, tracking and behaviour metric extraction using deep learning. *IEEE Access* **2019**, *7*, 108049–108060. [CrossRef]
45. Leichter, I.; Krupka, E. Monotonicity and error type differentiability in performance measures for target detection and tracking in video. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2553–2560. [CrossRef] [PubMed]
46. Luo, W.; Xing, J.; Milan, A.; Zhang, X.; Liu, W.; Kim, T.K. Multiple object tracking: A literature review. *Artif. Intell.* **2020**, *293*, 103448. [CrossRef]
47. Wang, A.; Sun, Y.; Kortylewski, A.; Yuille, A.L. Robust object detection under occlusion with context-aware compositionalnets. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 12645–12654.
48. Kortylewski, A.; Liu, Q.; Wang, A.; Sun, Y.; Yuille, A. Compositional convolutional neural networks: A robust and interpretable model for object recognition under occlusion. *Int. J. Comput. Vis.* **2021**, *129*, 736–760. [CrossRef]
49. Cosgrove, C.; Kortylewski, A.; Yang, C.; Yuille, A. Robustness Out of the Box: Compositional Representations Naturally Defend Against Black-Box Patch Attacks. *arXiv* **2020**, arXiv:2012.00558.
50. Kortylewski, A.; He, J.; Liu, Q.; Yuille, A.L. Compositional convolutional neural networks: A deep architecture with innate robustness to partial occlusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 8940–8949.
51. Jeong, J.M.; Yoon, T.S.; Park, J.B. Kalman filter based multiple objects detection-tracking algorithm robust to occlusion. In Proceedings of the 2014 Proceedings of the SICE Annual Conference (SICE), Sapporo, Japan, 9–12 September 2014; pp. 941–946.
52. Li, X.; Wang, K.; Wang, W.; Li, Y. A multiple object tracking method using Kalman filter. In Proceedings of the 2010 IEEE International Conference on Information and Automation, Harbin, China, 20–23 June 2010; pp. 1862–1866.
53. Hou, X.; Wang, Y.; Chau, L.P. Vehicle tracking using deep sort with low confidence track filtering. In Proceedings of the 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Taipei, Taiwan, 18–21 September 2019; pp. 1–6.
54. Frossard, D.; Urtasun, R. End-to-end learning of multi-sensor 3D tracking by detection. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 635–642.
55. Smith, J.E.; Pinter-Wollman, N. Observing the unwatchable: Integrating automated sensing, naturalistic observations and animal social network analysis in the age of big data. *J. Anim. Ecol.* **2021**, *90*, 62–75. [CrossRef]
56. Kakanis, M.; Sossidou, E.; Kritas, S.; Tzika, E. Update on Tail biting in pigs: An undesirable damaging behaviour. *J. Hell. Vet. Med Soc.* **2021**, *72*, 2629–2646. [CrossRef]
57. Larsen, M.L.V.; Pedersen, L.J.; Edwards, S.; Albanie, S.; Dawkins, M.S. Movement change detected by optical flow precedes, but does not predict, tail-biting in pigs. *Livest. Sci.* **2020**, *240*, 104136. [CrossRef]
58. D'Eath, R.B.; Jack, M.; Futro, A.; Talbot, D.; Zhu, Q.; Barclay, D.; Baxter, E.M. Automatic early warning of tail biting in pigs: 3D cameras can detect lowered tail posture before an outbreak. *PLoS ONE* **2018**, *13*, e0194524.
59. Liu, D.; Oczak, M.; Maschat, K.; Baumgartner, J.; Pletzer, B.; He, D.; Norton, T. A computer vision-based method for spatial-temporal action recognition of tail-biting behaviour in group-housed pigs. *Biosyst. Eng.* **2020**, *195*, 27–41. [CrossRef]
60. Ghaffarian, S.; Valente, J.; Van Der Voort, M.; Tekinerdogan, B. Effect of Attention Mechanism in Deep Learning-Based Remote Sensing Image Processing: A Systematic Literature Review. *Remote Sens.* **2021**, *13*, 2965. [CrossRef]
61. Xu, R.; Tao, Y.; Lu, Z.; Zhong, Y. Attention-mechanism-containing neural networks for high-resolution remote sensing image classification. *Remote Sens.* **2018**, *10*, 1602. [CrossRef]
62. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.